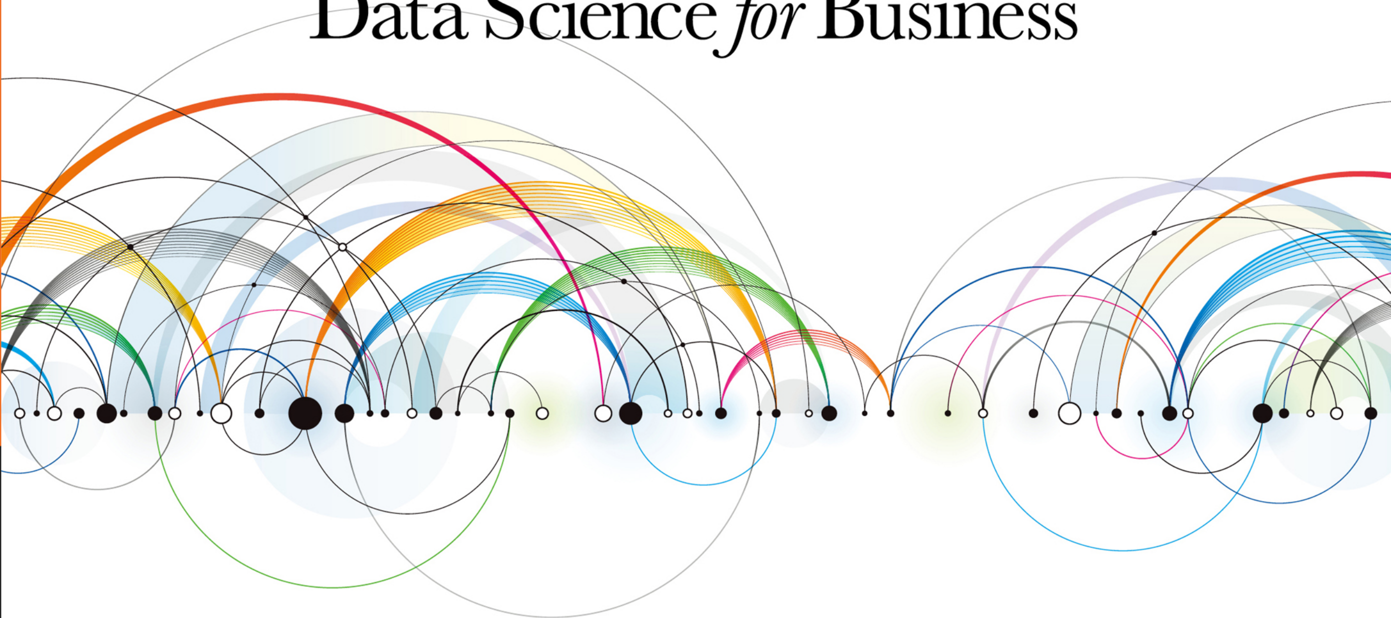
O'REILLY®

TURING 图灵程序设计丛书

商战数据挖掘

你需要了解的数据科学与分析思维

Data Science *for* Business



[美] 福斯特·普罗沃斯特 汤姆·福西特 著

郭鹏程 管晨 译

 中国工信出版集团  人民邮电出版社 POSTS & TELECOM PRESS

作者介绍

福斯特·普罗沃斯特（Foster Provost）



纽约大学斯特恩商学院教授，教授商业分析、数据科学与MBA课程。他曾是Verizon公司研究型数据科学家，参与创建过多家成功的数据科学驱动企业。

汤姆·福西特（Tom Fawcett）



机器学习博士，Data Science LLC首席数据科学家，从事应用机器学习研究和数据挖掘20余年，发表过大量机器学习文章。

译者介绍

郭鹏程



物理学出身，获天文学博士学位，是科普爱好者和数据科学家，当前致力于将数据科学应用到实际业务中，专门从事数据科学相关的培训、咨询、设计和开发等业务。目前任教于山东财经大学数学与数量经济学院。

管晨



毕业于山东财经大学金融数学系，擅长数据分析和数据可视化，也是语言爱好者，现从事商业地产和教育方面的商业智能工作。

数字版权声明



图灵程序设计丛书

商战数据挖掘：

你需要了解的数据科学与分析思维

Data Science for Business:

What You Need to Know about Data Mining and Data-Analytic Thinking

[美] 福斯特·普罗沃斯特　汤姆·福西特　著

郭鹏程　管晨　译

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

O'Reilly Media, Inc.授权人民邮电出版社出版

人民邮电出版社

北　京

图书在版编目（CIP）数据

商战数据挖掘 ： 你需要了解的数据科学与分析思维 / （美）福斯特・普罗沃斯特（Foster Provost），（美）汤姆・福西特（Tom Fawcett）著 ； 郭鹏程 管晨译. -- 北京 ： 人民邮电出版社，2019.12
（图灵程序设计丛书）
ISBN 978-7-115-52233-7

Ⅰ. ①商… Ⅱ. ①福… ②汤… ③郭… Ⅲ. ①数据采集 Ⅳ. ①TP274

中国版本图书馆CIP数据核字(2019)第219030号

内 容 提 要

数据挖掘是现代企业从数据中提取有用信息、获取竞争优势的重要方法。针对数据科学的这一商业应用，本书进行了深入解读，不仅详细介绍了数据挖掘的环节、常用分析技术和基本模型，还提供了数据科学解决方案的提案示例和评估指南。同时，为了便于读者理解，本书不仅分析了大量商业示例，在业务情景下阐释数据挖掘的基本概念和原理，还使用大量图表辅助解释数学细节。因此，读者无须专业数学背景即可阅读本书。

本书适合数据科学项目管理者、数据科学企业投资者、数据科学项目的开发者，以及其他有志于研究数据科学的人士阅读。

◆ 著　[美] 福斯特・普罗沃斯特　汤姆・福西特
译　郭鹏程　管　晨
责任编辑　岳新欣
责任印制　周昇亮

◆ 人民邮电出版社出版发行　北京市丰台区成寿寺路11号
邮编　100164　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京　印刷

◆ 开本：800×1000　1/16
印张：18.75
字数：443千字　2019年 12 月第 1 版
印数：1 – 3 500册　2019年 12 月北京第 1 次印刷

著作权合同登记号　图字：01-2019-6403号

定价：89.00元

读者服务热线：(010)51095183转600　印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

版权声明

O’Reilly Media, Inc.介绍

O’Reilly 以“分享创新知识、改变世界”为己任。40 多年来我们一直向企业、个人提供成功必需之技能及思想，激励他们创新并做得更好。

O’Reilly 业务的核心是独特的专家及创新者网络，他们通过我们分享知识。我们的在线学习（Online Learning）平台提供独家的直播培训、图书及视频，使客户更容易获取业务成功所需的专业知识。几十年来 O’Reilly 图书一直被视为学习开创未来之技术的权威资料。我们全年举办的诸多会议是活跃的技术聚会场所，来自各领域的专业人士在此建立联系，讨论最佳实践并发现可能影响技术行业未来的新趋势。

我们的客户渴望作出推动世界前进的创新，我们能祝您一臂之力。

业界评论

“O’Reilly Radar 博客有口皆碑。”

——*Wired*

“O’Reilly 凭借一系列非凡想法（真希望当初我也想到了）建立了数百万美元的业务。”

——*Business 2.0*

“O’Reilly Conference 是聚集关键思想领袖的绝对典范。”

——*CRN*

“一本 O’Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——*Irish Times*

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野，并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去，Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——*Linux Journal*

谨以此书献给我们的父亲们。

目录

赞誉 xiii

译者序 xv

前言 xvii

第 1 章　绪论：数据分析式思维 1

1.1　数据机遇无处不在 1
1.2　案例：飓风 Frances 2
1.3　案例：预测用户流失 3
1.4　数据科学、数据工程和数据驱动型决策 4
1.5　数据处理和“大数据” 6
1.6　从大数据 1.0 到大数据 2.0 6
1.7　数据与数据科学能力：一种战略性资产 7
1.8　数据分析式思维 9
1.9　关于本书 10
1.10　重新审视数据挖掘和数据科学 11
1.11　数据科学：一门新兴的实验性学科 12
1.12　小结 12

第 2 章　商业问题及其数据科学解决方案 14

2.1　从商业问题到数据挖掘任务 14
2.2　有监督方法与无监督方法 17
2.3　数据挖掘及其结果 18
2.4　数据挖掘流程 19

2.4.1 业务理解环节 …… 20
2.4.2 数据理解环节 …… 21
2.4.3 数据准备环节 …… 22
2.4.4 建模环节 …… 22
2.4.5 评估环节 …… 23
2.4.6 部署环节 …… 24
2.5 管理数据科学团队的含义 …… 25
2.6 其他分析技巧与技术 …… 26
2.6.1 统计 …… 26
2.6.2 数据库查询 …… 27
2.6.3 数据仓库 …… 28
2.6.4 回归分析 …… 28
2.6.5 机器学习与数据挖掘 …… 28
2.6.6 运用以上技术解决商业问题 …… 29
2.7 小结 …… 30

第 3 章 预测建模导论：从相关性到有监督的划分 …… 31

3.1 建模、归纳与预测 …… 32
3.2 有监督的划分 …… 35
3.2.1 选取富信息属性 …… 36
3.2.2 示例：基于信息增益进行属性选择 …… 42
3.2.3 使用树形结构模型进行有监督的划分 …… 46
3.3 划分的可视化 …… 52
3.4 把树视作规则组 …… 53
3.5 概率估计 …… 54
3.6 示例：用树型归纳解决用户流失问题 …… 56
3.7 小结 …… 59

第 4 章 用模型拟合数据 …… 61

4.1 根据数学函数分类 …… 62
4.1.1 线性判别函数 …… 64
4.1.2 目标函数的最优化 …… 66
4.1.3 示例：基于数据挖掘线性判别式 …… 67
4.1.4 用线性判别函数对实例进行评分和排序 …… 68
4.1.5 支持向量机简介 …… 69
4.2 通过数学函数进行回归 …… 71
4.3 类概率估计和逻辑“回归” …… 73
4.4 示例：对比逻辑回归和树型归纳 …… 77
4.5 非线性函数、支持向量机和神经网络 …… 81
4.6 小结 …… 83

第 5 章　避免过拟合 ······ 84
5.1　泛化能力 ······ 84
5.2　过拟合 ······ 85
5.3　过拟合检验 ······ 86
5.3.1　保留数据和拟合图 ······ 86
5.3.2　树型归纳的过拟合问题 ······ 88
5.3.3　数值函数的过拟合问题 ······ 89
5.4　示例：线性函数的过拟合 ······ 90
5.5　* 示例：过拟合为何有害 ······ 95
5.6　从保留评估到交叉验证 ······ 96
5.7　用户流失数据集回顾 ······ 99
5.8　学习曲线 ······ 100
5.9　避免过拟合与控制复杂度 ······ 101
5.9.1　树型归纳中的过拟合规避 ······ 102
5.9.2　避免过拟合的一般方法 ······ 102
5.9.3　* 参数优化中的过拟合规避 ······ 104
5.10　小结 ······ 106
第 6 章　相似性、近邻和簇 ······ 107
6.1　相似性和距离 ······ 108
6.2　最近邻推理 ······ 109
6.2.1　示例：威士忌分析 ······ 110
6.2.2　用最近邻来进行预测建模 ······ 111
6.2.3　近邻的数量及其影响 ······ 113
6.2.4　几何解释、过拟合和复杂度控制 ······ 115
6.2.5　最近邻方法的问题 ······ 118
6.3　与相似性和最近邻相关的一些重要技术细节 ······ 119
6.3.1　混合属性 ······ 119
6.3.2　* 其他距离函数 ······ 120
6.3.3　* 组合函数：计算近邻的评分 ······ 122
6.4　聚类 ······ 124
6.4.1　示例：威士忌分析回顾 ······ 124
6.4.2　层次聚类 ······ 125
6.4.3　最近邻回顾：根据形心的聚类 ······ 128
6.4.4　示例：对商业新闻报道进行聚类 ······ 132
6.4.5　理解聚类结果 ······ 135
6.4.6　* 用有监督学习产生簇描述 ······ 136
6.5　退一步：解决业务问题与数据探索 ······ 139
6.6　小结 ······ 140

第 7 章　决策分析思维（一）：如何评估一个模型 142
7.1 对分类器的评估 143
7.1.1 简单准确率的问题 143
7.1.2 混淆矩阵 144
7.1.3 样本类别不均衡的问题 144
7.1.4 成本收益不均衡的问题 147
7.2 分类问题的推广 147
7.3 一个重要的分析框架：期望值 148
7.3.1 用期望值规范分类器的使用 148
7.3.2 用期望值规范分类器的评估 149
7.4 评估、基线性能以及对数据投资的意义 155
7.5 小结 157
第 8 章　模型性能的可视化 159
8.1 排序，而不是分类 159
8.2 利润曲线 161
8.3 ROC 图像和曲线 163
8.4 ROC 曲线下面积 168
8.5 累积响应曲线和提升曲线 168
8.6 示例：用户流失模型的性能分析 171
8.7 小结 177
第 9 章　证据和概率 179
9.1 示例：向线上目标用户投放广告 179
9.2 根据概率合并证据 181
9.2.1 联合概率与独立性 181
9.2.2 贝叶斯法则 182
9.3 将贝叶斯法则应用到数据科学中 183
9.3.1 条件独立和朴素贝叶斯 184
9.3.2 朴素贝叶斯的优劣势 186
9.4 证据“提升度”的模型 187
9.5 示例：Facebook“点赞”的证据提升度 188
9.6 小结 190
第 10 章　文本的表示和挖掘 191
10.1 为什么文本很重要 192
10.2 为什么文本很难处理 192
10.3 表示法 193
10.3.1 词袋模型 193

10.3.2 词频 ······193
10.3.3 度量稀疏度：逆文档频率 ······195
10.3.4 TFIDF ······196
10.4 示例：爵士音乐家 ······197
10.5 *IDF 和熵的关系 ······200
10.6 词袋模型之外的方法 ······202
10.6.1 n-grams 序列 ······202
10.6.2 命名实体提取 ······202
10.6.3 主题模型 ······203
10.7 示例：通过挖掘新闻报道预测股价变动 ······204
10.7.1 任务 ······204
10.7.2 数据 ······205
10.7.3 数据处理 ······207
10.7.4 结果 ······208
10.8 小结 ······211

第 11 章 决策分析思维（二）：面向分析工程 ······212

11.1 为慈善机构寻找最佳捐赠人 ······213
11.1.1 期望值框架：分解商业问题，重组解决方案 ······213
11.1.2 简短的题外话：选择性偏差 ······214
11.2 更复杂的用户流失示例回顾 ······215
11.2.1 期望值框架：构建更复杂的商业问题 ······215
11.2.2 评估激励的影响 ······216
11.2.3 从期望值分解到数据科学解决方案 ······217
11.3 小结 ······219

第 12 章 其他数据科学任务与技术 ······220

12.1 共现和关联：寻找匹配项 ······221
12.1.1 度量意外：提升度和杠杆率 ······221
12.1.2 示例：啤酒和彩票 ······222
12.1.3 Facebook 点赞的关联 ······223
12.2 用户画像：寻找典型行为 ······225
12.3 链路预测和社交推荐 ······229
12.4 数据约简、潜在信息和电影推荐 ······230
12.5 偏差、方差和集成方法 ······233
12.6 数据驱动的因果解释和一个病毒式营销示例 ······235
12.7 小结 ······236

第 13 章 数据科学和经营战略 ······237

13.1 数据分析式思维，终极版 ······237

13.2 用数据科学取得竞争优势……238
13.3 用数据科学保持竞争优势……239
13.3.1 令人敬畏的历史优势……240
13.3.2 独一无二的知识产权……240
13.3.3 独一无二的无形抵押资产……240
13.3.4 优秀的数据科学家……241
13.3.5 优秀的数据科学管理……242
13.4 吸引和培养数据科学家及其团队……243
13.5 检验数据科学案例分析……244
13.6 做好准备，接受来源各异的创意……245
13.7 做好准备，评估数据科学项目提案……245
13.7.1 数据挖掘提案示例……246
13.7.2 Big Red 提案中的缺陷……246
13.8 企业的数据科学成熟度……247
第 14 章 总结……250
14.1 数据科学的基本概念……250
14.1.1 将基本概念应用于新问题：挖掘移动设备数据……252
14.1.2 改变对商业问题解决方案的思考方式……253
14.2 数据做不到的：圈中人回顾……254
14.3 隐私、道德和挖掘个人数据……256
14.4 数据科学是否还有更多内容……257
14.5 最后一例：从众包到云包……257
14.6 最后的话……258
附录 A 提案评估指南……259
附录 B 另一个提案示例……262
术语表……265
参考文献……270
关于作者……278

赞誉

“对于每一个真诚拥抱大数据机遇的人来说，这都是一本必读之书。”

——Craig Vaughan，SAP 全球副总裁

“这本书适时地宣告了一个事实：在现代社会中，数据即商业，两者无法割裂。阅读这本书，你将理解数据思维背后的科学。”

——Ron Bekkerman，Carmel Ventures 首席数据官

“对于领导数据科学家或与之交互的商业管理者而言，如果希望略过教科书中的技术细节，而深入理解数据科学的原理和算法，这本书是绝佳选择。”

——Ronny Kohavi，Microsoft 在线服务部架构师

“Provost 和 Fawcett 萃取当今现实世界数据分析的艺术与科学之精华，汇集成了这本数据领域无与伦比的入门之作。”

——Geoff Webb，*Data Mining and Knowledge Discovery* 总编辑

“我希望所有与我共事的人都读过这本书。”

——Claudia Perlich，Dstillery 首席科学家，2013 年广告研究基金会创新奖获得者

“这本书是飞速发展的大数据领域之基石，是所有对大数据革命感兴趣之人的必读之物。”

——Justin Gapper，Teledyne Scientific and Imaging 业务部门分析经理

“两名作者都是在‘数据科学’还没有规范名称前就颇具成就的专家，他们将这个复杂的话题讲得浅显易懂，这对初级数据科学家尤其有益。这本书关注数据科学概念在实际商业问题中的应用，据我所知，它是首本涉及这一主题的书。书中大量引用了反映商业中常见问题的现实案例，如用户流失、目标市场营销，甚至威士忌的分析，这些案例极具说服力。

“这本书独一无二，因为它没有详解算法，而是帮读者理解数据科学背后的基本概念，最重要的是，它指导读者如何着手解决问题并取得成功。无论是想综合了解数据科学的普通人，还是需要学习基础知识的初级数据科学家，都要读一读这本书。”

——Chris Volinsky，AT&T 实验室统计研究主管，Netflix 百万美元挑战赛优胜组成员

“这本书远不止是数据分析入门书，对所有需要做出数据驱动型决策的人来说，这本书堪称必备指南。”

——Tom Phillips，Dstillery 首席执行官，Google 搜索和分析前主管

“善用数据已成为提升行业竞争力的一种强大力量。要想在这个数据驱动的环境中发展，数据工程师、分析师和经理等都必须理解其面临的选择、设计选项和利弊。本书案例有趣、叙述清晰，不仅细致地说明了‘怎么做’，也解释了‘为什么’。对于意图在数据驱动系统的发展和应用中有所作为的读者而言，这是一本完美的入门书。”

——Josh Attenberg，Etsy 数据科学负责人

“数据是生产率增长、创新以及深刻用户洞见的基础。善用数据的能力直到最近才被广泛视为竞争优势，并迅速成为在商业中立于不败之地的筹码。两位作者凭借丰富的应用经验，让这本书成为不二之选——它打开了一扇可以洞悉竞争对手策略的窗户。”

——Alan Murray，连续创业者，Coriolis Ventures 合伙人

“这是最好的数据挖掘书之一，让我彻底明白了外汇中流动性分析的相关概念。书中的例子非常恰当，能帮你深入理解这个主题。这本书将成为我的常备参考书。”

——Nidhi Kathuria，苏格兰皇家银行外汇副总裁

“这是一本绝佳的、通俗易懂的入门读物，它既能帮助商务人士更好地领会数据科学家所用的概念、工具和技术，又能帮助数据科学家更好地理解其解决方案所应用的商业背景。”

——Joe McCarthy，Atigeo 分析与数据科学主管

“我认为，对于必须在现实世界中应用数据科学和大数据技术的商业分析师和管理者来说，这本书是掌握这些技术的最佳选择。”

——Ira Laefsky，工程学硕士（计算机科学）/ 信息技术 MBA，人机交互研究员，
前 Arthur D. Little 和数字设备公司高级顾问

译者序

不同于其他讲述数据科学的书，本书从非数据科学人员，也就是管理者、投资者甚至工程师等人员的角度，阐述了数据科学这一新兴行业（或学科）的基本原理和基础理念，而这正是本书的惊艳之处。

作为一名数据科学工作者，身处数据科学快速发展的浪潮之中，我近年来参与了多家企业的数据项目。这些企业中虽然很少有像阿里、百度、电信那样的超大规模公司，却不乏经营了十几年或信息化多年的老牌企业，而这些企业希望利用积累多年的经营数据来实现精细化经营。此外，也有很多不同行业的创业公司希望将大数据分析和挖掘作为契机来“撬开”市场。他们（包括我自己）最常遇到的问题，就是难以正确地评估数据的成本和价值以及恰如其分地把握数据项目的路径和节奏。当“商业智能”“大数据”“数据挖掘”“数据分析”“智慧城市”“智能运营”“增长黑客”“机器学习”“深度学习”“人工智能”等热门词语轮番被媒体和业界追捧的时候，技术人员关心如何快速地“掌握”算法包从而提高薪酬，经营者关注如何搭上热点的快车，却很少有人冷静地分析这些热点背后的实质——数据科学。2018 年，我翻译本书之际，正值信息技术产业遭遇寒流，很多创业公司（特别是一些“数据”“智能”公司）停滞甚至关闭，大量创投遇冷，而本书所述的数据科学的原理和理念或许可以帮助我们理解、反思这些现象。

试举两个例子。一家号称经营“能源智能运维”的企业积累了很多设备数据，希望以此构建故障的预测算法，进而实现提前备件和维护的能力。但是，我们在评估这些数据时，却发现其中并没有关于“故障”的清晰、准确的记录。于是，我们告知企业的管理者他们缺乏有效标注数据（相关概念可以参见本书第 3 章），希望他们能够改善数据积累流程，也就是智能运维的数据收集机制。但是该企业坚持认为数据已经足够多（实际上，数万台设备的秒级数据，量的确很大），没有接受我们的建议。目前这家企业已经转型做施工了。

另一个案例和一家上市公司与政府合作的 PPP 新项目有关。上市公司打出大数据驱动的旗号，并声称他们将“整合行业资源，利用数据为行业‘赋能’”，一时间备受瞩目。然而该项目有一个最大的问题：没有数据。在项目筹建初期，决策层为了能“漂亮”地亮相，将本应用于工程和数据团队的预算用在了装修和高价购买数据上。因此，系统虽然“上线”

了，但是其中的数据是“死”的。正是由于上市公司没能正确地评估数据的价值，积极寻求数据路径（相关内容参见本书第 1、2、13 章及附录），一年之后，该项目依然没有稳定的数据源。所幸，经过一番人事更迭，两三年后该项目重回起点，踏踏实实地从头开始运营，目前已经颇具名气。

本书第 3~12 章虽然讲了若干基本的数据科学方法，但是视角颇为独特。本书按照方法的基本原理而非功能（例如回归分析、分类分析、聚类分析、关联分析等经典归类法）来归类。以我的理解，这是根据数据所蕴含的信息量进行的分类。书中不仅很少有公式，甚至一行代码也没有出现，跟任何编程语言都无关。这绝非刻意迎合非技术背景的读者，而是因为阐述数据科学的原理和理念本就不需要任何代码，少量的公式只是为了帮助读者了解**确实存在**一些确定的方法来量化地表示那些看似模糊的概念（如信息量）。

书中第 13 章所述案例恰当地指出了数据团队和经营管理者之间沟通的障碍所在。作者显然也受过不少“委屈”，书中描述的一些情景似曾相识，让我在翻译过程中哑然失笑。但是，这些障碍不能只归咎于管理人员。第 7、8、11 章中介绍的一些评估方法，让我能更多地从经营者和管理人员的视角看待数据问题，因此本书除了面向非数据科学背景的读者，也绝对适合数据人员。它有助于降低数据团队内部沟通以及团队与外部沟通的成本，从而提升数据团队的价值。

数据科学本身并不是一个非常新的行业或学科。早在 20 世纪，一些美国电影中就出现过依据数据进行决策的桥段。只不过它在众多耀眼的近义词的喧嚣中显得很普通。本书讲述了数据科学的原理和方法，并特别强调了 CRISP-DM（跨行业数据挖掘标准流程），该流程可以帮助数据项目建立合理的路径和里程碑，有效控制数据项目的风险。同时本书向我们传达了数个有关数据科学的理念，例如：“数据和数据分析能力应被视为企业的资产而非成本”。

本书的另一位译者，管晨女士，是我曾经的学生，也是我现在从事数据驱动运营的同事，我们共同翻译了本书的每一章。此外，我还要特别感谢王大鹏和张国文在本书翻译过程中提供的重要意见和建议。两位都曾上过我的课，现在大鹏是我的同事，也是数据挖掘方面的专家，而国文也活跃在咨询行业的多个数字化转型项目中。

图灵公司的图书充满了科技气息，是我的最爱，我非常荣幸有机会参与到图灵公司的图书出版中。最后，特别感谢图灵公司的编辑朱巍、岳新欣和祁玥以及幕后很多我还不知晓姓名的编辑老师们，他们的辛勤工作和严格要求保障了本书的翻译质量。

郭鹏程
2019 年 9 月
于山东济南

前言

本书适合以下几种读者：

- 准备与数据科学家合作、管理面向数据科学的项目或投资数据科学企业的商业人士；
- 即将实施数据科学解决方案的开发人员；
- 志向远大的数据科学家。

本书不讨论算法，不能取代算法主题的图书。我们故意没有采用以算法为中心的方法，是因为我们相信，在从数据中提取有用信息的技术的背后，存在着一套精简的基本概念或原理，而它们构成了许多著名的数据挖掘算法的基础。此外，它们还支撑着以数据为中心的行业问题的分析、数据科学解决方案的构建和评估，以及一般性数据科学策略和提案的评估。因此，我们围绕这些一般性概念和原理而非具体算法组织了本书内容。当有必要描述程序细节时，本书会用文字和图表相结合的方式进行解释，因为我们认为这样比列出详细的算法步骤更易于理解。

尽管本书不要求读者有专业的数学背景，但本书内容具有一定的技术性——本书旨在让读者深入理解数据科学，而非仅对其有个大体认识。本书尽量少用数学语言，多做概念性阐述。

业界同行评价说，本书是能帮助业务团队、技术 / 研发团队和数据科学团队形成统一认识的无价之宝。这个结论是从一小部分人身上得出的，而我们想知道本书的适用范围到底有多广（详见第 5 章）。我们希望每位数据科学家都能把这本书推荐给其开发团队或者业务团队里的同事，并对他们说“如果你们迫切希望通过设计 / 实施顶尖的数据科学解决方案来解决商业问题，那么我们必须对这本书的内容有共同的理解”。

同行还告诉我们，这本书还有个意料之外的作用：可以用来准备数据科学类职位的面试。企业对数据科学家的需求日益增长，相应地，越来越多的求职者自称是数据科学家。每个数据科学岗位的求职者都需要理解本书中呈现的基本原理。（业界同行说，他们很惊讶竟然有那么多人做不到这一点。我们甚至半开玩笑地讨论，是不是紧接着写一本《数据科学求职者笔记》。）

学习数据科学的概念性方法

本书介绍了数据科学中最重要的基本概念。其中一些概念直接体现在了章名中，其他的则会在讨论过程中自然而然地呈现出来（因此不会被标注为"基本概念"）。这些概念贯穿整个学习过程，从构想问题到应用数据科学方法，再到运用结果改进决策。同时，它们也构成了大量商业分析方法与技术的基础。

这些概念主要分为以下三种。

(1) 关于数据科学如何融入组织和竞争环境的概念，包括如何吸引、组织和培养数据科学团队，如何让数据科学转化为竞争优势，以及如何做好数据科学项目。

(2) 形成数据分析式思维的一般方法。它们有助于识别合适的数据，选择合适的方法。这些概念包括**数据挖掘过程**和一系列不同的**高级数据挖掘任务**。

(3) 从数据中获取信息的一般性概念。这些概念为大量的数据科学任务及其算法奠定了基础。

比如，有一条基本概念是如何判定两个由数据描述的个体之间的相似性。这项能力是执行多种具体任务的基础：它可以直接用于**寻找**与指定用户相似的用户；它构成了许多**预测**算法的核心，可以用来估计目标值，如资源使用量或用户响应促销活动的概率；它还是**聚类**方法的基础，即在没有特定目标的情况下，按照个体之间共有的特征将它们分组。相似性同样是**信息检索**的基础，可以检索出一系列与查询词条相关的文件或网页。最后，它也是许多**推荐**算法的基础。在传统的面向算法的书中，这些任务可能会以不同的名字分布于不同章节，其中的共同点却被掩盖在重重的算法与数学命题之下。本书关注的是统一的概念，而各个具体的任务和算法就是它们的自然呈现。

再举一个例子，在对模式的效用进行评估时，**提升度**（lift）这一指标在数据科学领域随处可见。它指的是某一模式在多大程度上是由非随机情况导致的。它可以用于在不同场景下对不同模式进行评估。例如，通过计算目标人群的提升度，可以评估定向广告算法。它还可以用于判断结论的正负证据权重（WOE），以及判断数据中的共现情况是否有意义，不同于仅是高频事件的自然结果。

我们相信，运用这些基本概念来解释数据科学，不仅能帮助读者学习，还能促进企业利益相关者与数据科学家之间的交流。这种方式使得双方语言共通，从而能更好地理解彼此。而概念共通又能让各方进行更深入的讨论，从而发现之前可能被忽略的关键问题。

写给教师们

本书被许多数据科学课程用作教材，而且颇为成功。本书最初的灵感来源于 2005 年秋季 Foster 在纽约大学斯特恩商学院开设的跨学科数据科学课程。[1] 尽管最初这门课程是为 MBA（工商管理硕士）和 MSIS（信息系统硕士）开设的，却吸引了校内各个学科的学生。这门课最有趣的地方不是它吸引了 MBA 和 MSIS——原本就是为他们开设的，而是它对有机器学习和其他技术类学科背景的学生同样非常有价值。我们猜想，部分原因大概是他们的课程仅聚焦在算法上，缺失了基本原理和其他内容。

注 1：当然了，一本书的每个作者都认为自己做的贡献更多。

目前，纽约大学用这本书来辅助众多与数据科学相关的教学项目，如最初的 MBA 和 MSIS 项目、本科商业分析课程、斯特恩商学院的商业分析硕士项目，以及纽约大学最新的数据科学硕士项目中的数据科学入门课程。此外，本书（出版前）已被 9 个国家的 20 余所高校采用（数目仍在增长），用于商学院、计算机科学项目和数据科学初级课程。

其他技能及概念

除了数据科学的基本原理外，实干的数据科学家还需要了解和掌握许多其他概念与技能，这些会在第 1 章和第 2 章讲到。

本书结构及体例

除了偶尔出现的脚注，本书还会出现用方框框起的“补充栏”。它们本质上是扩展了的脚注，用于阐释那些有趣、有价值，但作为脚注过长，又偏离主题的内容。



前方有技术细节 —— 关于带星号的小节的说明

我们把偶尔出现的数学细节归入了带星号的选读小节中。这些小节的标题前带有星号，小节开头还有这样的一段辅文。这些小节包含更多的数学 / 技术细节，这段文字就解释了其目的。读者在阅读本书时，即使跳过这些部分也不会影响阅读的连续性，但本书仍会在一些地方提醒读者该处将介绍技术细节。

本书中如“（Smith & Jones, 2003）”这样的文本表示对参考文献中一个条目的引用（此处即指，Smith 和 Jones 在 2003 年发表的文章或出版的图书）；“Smith & Jones（2003）”与之类似。全书使用的参考文献列在正文后面。

本书尽可能少讲数学，并且在讲到数学的时候进行了简化，以免造成困惑。针对有技术背景的读者，我们有必要对简化方式稍作解释。

(1) 我们没有使用教科书中普遍使用的 Sigma（Σ，连加）和 Pi（Π，连乘）符号，而是使用了如下带省略号的公式：

$$f(x) = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$$

在介绍技术细节的带星号小节中，若上述方法过于繁冗，本书偶尔也会使用 Σ 和 Π 符号。我们假定阅读这些部分的读者习惯使用这种符号，不会感到困惑。

(2) 统计学图书通常会在估计值上加上“帽子”符号，以区分真实值和其估计值，因此在这类书中，你往往会看到实际概率表示为 p，而其估计值表示为 $\hat{p}$。本书几乎一直讨论基于数据的估计值，加上帽子符号会让公式又复杂又难看，因此除非特别指出，否则这些值默认都是基于数据的估计值。

(3) 一些符号和变量在上下文中不言自明，因此我们会在文中简化或删除它们。比如，在用数学语言讨论分类器时，技术上讲，本书表示的是基于特征向量所进行的决策预测。以较为正式的方式表示，就会得到如下公式：

$$\hat{f}_R(\boldsymbol{x}) = x_{\text{Age}} \times (-1) + 0.7 \times x_{\text{Balance}} + 60$$

其中，Age 表示年龄，Balance 表示账户余额。但我们把它写得更通俗易懂：

$$f(\boldsymbol{x}) = \text{Age} \times (-1) + 0.7 \times \text{Balance} + 60$$

其中，$\boldsymbol{x}$ 是向量，Age 和 Balance 是向量的元素。

为了尽量保持版式一致，本书用等宽字体（如 `sepal_width`）表示数据中的属性或关键字。比如，在第 10 章中，`discuss` 表示数据中的一个输出标记。

本书采用了如下排版约定。

- **黑体字**
  表示新术语或重点强调的内容。
- 等宽字体（`constant width`）
  表示程序片段，以及正文中出现的变量、函数名、数据库、数据类型、环境变量、语句和关键字等。
- 等宽斜体（*`constant width italic`*）
  表示应该由用户输入的值或根据上下文确定的值替换的文本。

本书中，我们在正文中穿插了一些与内容相关的提示和警告。根据阅读载体（纸质书、PDF 或电子书）的不同，它们的呈现形式会不大一样，如下所示。



该图标表示提示或建议。



该图标表示一般注解。



该图标表示警告或警示。它比提示重要得多，且出现得较少。

示例的使用

本书除了作为数据科学的入门读物，对在日常工作中进行探讨也颇有帮助。引用本书中的示例来回答问题无须获得许可。我们很希望但并不强制要求你在引用本书内容时加上引用说明。引用说明一般包括书名、作者、出版社和 ISBN。比如：“*Data Science for Business* by Foster Provost and Tom Fawcett (O’Reilly). Copyright 2013 Foster Provost and Tom Fawcett, 978-1-449-36132-7.”

如果你觉得自己对示例的使用超出了合理使用或上述许可的范围，请通过 permissions@oreilly.com 联系我们。

Safari® Books Online



Safari Books Online 是应运而生的数字图书馆。它同时以图书和视频的形式出版世界顶级技术和商务作家的专业作品。技术专家、软件开发人员、Web 设计师、商务人士和创意专家等，在开展调研、解决问题、学习和认证培训时，都将 Safari Books Online 视作获取资料的首选渠道。

对于组织团体、政府机构和个人，Safari Books Online 提供各种产品组合和灵活的定价策略。用户可通过一个功能完备的数据库检索系统访问 O'Reilly Media、Prentice Hall Professional、Addison-Wesley Professional、Microsoft Press、Sams、Que、Peachpit Press、Focal Press、Cisco Press、John Wiley & Sons、Syngress、Morgan Kaufmann、IBM Redbooks、Packt、Adobe Press、FT Press、Apress、Manning、New Riders、McGraw-Hill、Jones & Bartlett、Course Technology 以及其他几十家出版社的上千种图书、培训视频和正式出版之前的书稿。要了解 Safari Books Online 的更多信息，我们网上见。

联系我们

请把对本书的评价和问题发给出版社。

美国：

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472

中国：

北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室（100035）
奥莱利技术咨询（北京）有限公司

对于本书的评论和技术性问题，请发送电子邮件到：bookquestions@oreilly.com

要了解更多 O'Reilly 图书、培训课程、会议和新闻的信息，请访问以下网站：
http://www.oreilly.com

我们在 Facebook 的地址如下：http://facebook.com/oreilly

请关注我们的 Twitter 动态：http://twitter.com/oreillymedia

我们的 YouTube 视频地址如下：http://www.youtube.com/oreillymedia

致谢

感谢在与我们讨论或阅读手稿后，为我们提供宝贵思路、反馈、意见、建议和鼓励的所有同事和其他人。虽然可能有所遗漏，但我们想在此一一感谢：Panos Adamopoulos、Manuel Arriaga、Josh Attenberg、Solon Barocas、Ron Bekkerman、Josh Blumenstock、Ohad Brazilay、Aaron Brick、Jessica Clark、Nitesh Chawla、Peter Devito、Vasant Dhar、

Jan Ehmke、Theos Evgeniou、Justin Gapper、Tomer Geva、Daniel Gillick、Shawndra Hill、Nidhi Kathuria、Ronny Kohavi、Marios Kokkodis、Tom Lee、Philipp Marek、David Martens、Sophie Mohin、Lauren Moores、Alan Murray、Nick Nishimura、Balaji Padmanabhan、Jason Pan、Claudia Perlich、Gregory Piatetsky-Shapiro、Tom Phillips、Kevin Reilly、Maytal Saar-Tsechansky、Evan Sadler、Galit Shmueli、Roger Stein、Nick Street、Kiril Tsemekhman、Craig Vaughan、Chris Volinsky、Wally Wang、Geoff Webb、Debbie Yuster 以及 Rong Zheng。我们还想感谢 Foster 课上的同学们，这些课程包括商业分析的数据挖掘、实用数据科学、数据科学导论、数据科学研讨班。你们在使用本书早期手稿期间提出的相关问题，为本书的改进提供了重要参考。

感谢这些年来所有教过我们数据科学知识和数据科学教学方法的同事，尤其是 Maytal Saar-Tsechansky 和 Claudia Perlich。Maytal 曾在多年前慷慨地与 Foster 分享了她的数据挖掘课笔记。本书中的很多示例都基于她的思路和案例，比如第 3 章的分类树案例（尤其是“主体”可视化的部分），第 4 章中以可视化方式比较树模型和线性判别函数对实例空间的分割，第 6 章的“David 会响应吗”案例，等等。Claudia 过去几年曾经与 Foster 同期讲授过商业分析的数据挖掘以及数据科学导论课程，Foster 也从 Claudia 那里受益良多。

感谢 David Stillwell、Thore Graepel 和 Michael Kosinski 为书中的一些案例提供 Facebook 上的点赞数据。感谢 Nick Street 为我们提供细胞核数据，并允许我们在第 4 章中使用细胞核图像。感谢 David Martens 在手机定位可视化方面提供的帮助。感谢 Chris Volinsky 提供他在 Netflix 挑战赛中作品的数据。感谢 Sonny Tambe 很早就为我们提供了他在大数据技术与生产力方面的成果。感谢 Patrick Perry 为我们提供了第 12 章使用的银行电话中心案例。感谢 Geoff Webb 允许我们使用 Magnum Opus 关联挖掘系统。

最重要的是要感谢我们的家人，感谢他们给予我们的耐心、鼓励和爱。

我们在撰写本书时，使用了大量的开源软件及其案例。因此，我们还需感谢以下软件和程序包的开发者和贡献者：

- Python 和 Perl
- SciPy、NumPy、Matplotlib 和 Scikit-Learn
- Weka
- 加利福尼亚大学欧文分校的机器学习仓库（Bache & Lichman, 2013）

电子版

扫描如下二维码，即可购买本书电子版。



第1章

绪论：数据分析式思维

不要做渺小的梦，因为它们没有撼动人心的力量。

——歌德

在过去的十五年中，各企业在商业基础设施上大量投入，因此具备了更好的数据收集能力。如今，几乎每个商业环节都可以收集数据，有些环节甚至装备了专供数据收集之用的设备，比如运营管理、生产制造、供应链管理、用户行为、市场营销和工作流管理等环节。与此同时，外部数据，如市场趋势、业界新闻和竞争对手的一举一动等，可以通过互联网获得。在此背景下，人们自然更有兴趣从丰富的数据中获取有用的信息和知识——这恰好就是“数据科学”所特指的领域。

1.1 数据机遇无处不在

当大量的数据触手可及时，几乎各行各业的公司都关注通过数据开发来获得竞争优势。过去，公司可以聘用统计学家、建模工程师和分析师，组队对数据进行人工分析。然而，当今的数据量和复杂度已远远超出人工分析的能力范围。与此同时，随着计算机和互联网的普及以及其算力的增强，覆盖多种数据集的分析方法和挖掘算法不断被开发出来，使得数据分析的深度和广度达到了前所未有的程度。这些现象的集中出现，使得数据科学原理和数据挖掘技术在商业领域的应用变得越来越广泛。

数据挖掘技术最常见的应用是在营销领域，尤其是在目标市场营销、线上广告和交叉销售的推荐系统中。一般客户关系管理系统使用数据挖掘技术来分析客户行为，以提高客户留存率和最大化客户价值。金融业使用数据挖掘技术来进行信用评分和量化交易，并在运营中用它检测欺诈行为和优化生产资源。亚马逊和沃尔玛等大型零售商在其经营的各个环

节——从市场营销到供应链管理——都使用了数据挖掘技术。很多公司由于战略性地应用了数据科学，因而在市场中崭露头角，有的甚至变成了数据挖掘公司。

本书的首要目标是帮助读者从数据的角度看待商业问题，并从原理上理解如何从数据中获取有用的信息和知识（即建立数据分析式思维）。数据分析式思维包含一个基础架构和一套基本原理，理解它们至关重要。诚然，解决某些具体问题时，人们需要具备直觉、创意、常识以及领域知识。但数据视角可以提供一个基于上述架构和原理的框架，来系统地分析这些问题。这样，你在逐渐熟悉这种数据分析式思维之后，就会自然地培养出一种直觉，懂得在何处以何种方式运用你的创意和领域知识（这样的好处显而易见，因为宝贵的创意和知识需要用在最需要的地方）。

本书的第 1 章和第 2 章将详细讨论与数据科学和数据挖掘相关的多个话题和技术。本书会频繁使用"数据科学"和"数据挖掘"这两个术语，两者在很多情况下是可以混用的，不过"数据科学"这个字眼在各种以获利为目的的炒作中已经失去了它本来的意义。严格地说，"数据科学"是一套指导人们从数据中获取知识的基本原理，而"数据挖掘"则是将这些原理以具体技术的形式实现并从数据中获取知识的过程。作为术语，"数据科学"比传统意义上的"数据挖掘"涵盖的范围更广，而后者则对前者的原理进行了最清晰的阐释。



即使你没有任何亲自应用数据科学的打算，理解数据科学也是至关重要的。这是因为数据分析式思维可以帮助你评估与数据挖掘有关的商业提案。譬如当你的一位员工、一位咨询师或者一个潜在的投资对象提议通过对数据进行分析和挖掘来改善某一商业环节时，你应该有能力系统地评估该提案，判断它是否可行。当然，这并不意味着让你判断它是否一定会成功，因为"尝试"是数据挖掘项目的家常便饭，不成功的风险总是存在。但是至少你应该有能力发现一个提案是否存在明显的缺陷、不现实的假设或者缺失的环节。

本书将介绍大量的数据科学基础原理，同时每一条原理都会通过列举至少一项应用了该原理的数据挖掘技术来解释。由于每一条原理都会对应多项技术，因此本书把重点放在原理解释而非具体技术应用上。换言之，除非对理解概念有关键作用，否则本书不会大费周章地区分"数据科学"和"数据挖掘"这两个概念。

让我们来看两个通过分析数据发现预测性模式的简单案例。

1.2 案例：飓风Frances

2004 年，《纽约时报》刊登了这样一则报道：

> 飓风 Frances 正快速穿越加勒比海，并将直击佛罗里达洲东海岸地区。当地居民忙着前往海拔较高的地方避灾，而远在阿肯色州本顿维尔市的沃尔玛管理层却把这场灾害视为一个绝佳的机会，并计划借此展示他们最新的数据驱动法宝——预测技术。

> 飓风登陆前一周，沃尔玛首席信息官 Linda M. Dillman 让员工们根据数周前飓风 Charley 袭击的影响设计并开发出一套预测系统。依靠沃尔玛数据库中数万亿字节的客户消费记录数据，Linda 认为公司可以“化被动为主动，预测会发生什么事，而不是等着事情发生”。（Hays, 2004）

现在，思考一下，**为什么**数据驱动型预测在这种情况下能够派上用场。它也许能预测出飓风路线上的居民会需要更多的瓶装水。可这太显而易见了吧？即使不使用数据科学，我们也能知道。也许它能计算出飓风引起的瓶装水**销售增量**，进而保障飓风路线上的沃尔玛店有不多不少的库存。也许通过挖掘数据，可以发现在飓风路线上的沃尔玛商店里，某种 DVD 脱销了。但是有可能在那一周内，该 DVD 在全国所有的沃尔玛商店里都脱销了，而非仅限于那些飓风经过的沃尔玛店。数据驱动型预测或许多少有些作用，但是它的应用范围很可能比 Linda M. Dillman 最初计划的要更加广泛。

更有价值的是，数据驱动型预测可以用来发现在飓风影响下产生的隐含模式。为了做到这一点，分析师可能需要分析沃尔玛在相似情况下（比如数周前飓风 Charley 登陆期间）的海量数据，从中识别出当地不同寻常的产品需求。通过这样的一些模式，沃尔玛就能在飓风登陆之前预测到特殊的产品需求，并迅速补充相应库存。

实际上，这种情况真的发生了。《纽约时报》写道：“……专家在挖掘数据之后发现，除了那些常规的应急物资，某些特定商品的销量出人意料地增加了。‘我们之前从没想到，飓风到来前，草莓馅饼的销量会涨到平时的 7 倍！’Dillman 在采访中透露，‘而且销售冠军居然是啤酒。’”[1]

1.3　案例：预测用户流失

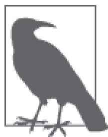
这类数据分析的效果如何？现在再来看一个更典型的商业案例，并审视该如何从数据视角思考商业问题。这个案例将在本书中反复出现，我们把它作为一个通用的参考例子，以便更好地阐明本书中的一些问题。

假设你刚在美国最大的一家电信公司 MegaTelCo 找到一份不错的分析师工作，然而公司目前正面临着严峻的无线业务用户流失问题。比如在美国东海岸中部，20% 的手机用户在合约到期后选择不再续约，而获得新用户却变得越来越艰难。由于手机市场已经饱和，因而曾经呈井喷式增长的无线业务如今也已势微。各家电信公司正在为了争夺对方的用户和留存自己的老用户斗得头破血流。“用户流失”是指用户未能留存在一个公司而转移至对手公司的情形。这种情形背后的代价是巨大的：用户转入的公司需要花大价钱才能吸引用户，而失去用户的公司也会损失收益。

分析并解决上述难题，就是你需要做的工作。因为吸引新用户比留存老用户的成本高得多，所以大部分预算应该用于留存老用户。市场部门已经制订了一份给留存用户的优惠方案，你的工作就是设计出一份精确、具体的计划，告诉数据部门如何依靠 MegaTelCo 庞大的数据资源，找出哪些用户最应该得到上述优惠，从而有效地防止这些用户在合约到期后流失。

注 1：当然啦，冰镇啤酒和草莓馅饼更配哦！

仔细想想：你会用到什么数据？又该怎么运用这些数据？尤其是在留存用户的奖励方案的预算已经确定的条件下，如何挑选一批特定用户，才能使公司的用户流失率达到最低？实际上，这个问题比看上去难得多。这个案例会在本书中被反复讨论，并且，随着你对数据科学的理解逐步加深，本书对这个问题的解答也会逐步深入。



现实中，用户留存是数据挖掘技术的主要应用方向之一，尤其是在电信业和金融业。这些行业通常也是使用数据挖掘技术最早和最广泛的，之后本书会讨论其原因。

1.4 数据科学、数据工程和数据驱动型决策

数据科学涉及从数据的自动化分析结果中理解现象的原理、过程和技巧。在商业领域，人们最关注的是如何改进决策过程，这也是数据科学的终极目标。因此，本书将侧重于讲解这一点。

图 1-1 把数据科学置于组织中其他过程之间，这些过程与数据相关且联系密切。该图将数据科学同其他在商业中日渐受到关注的数据处理过程区分开来。让我们从图中的最上部开始讨论。

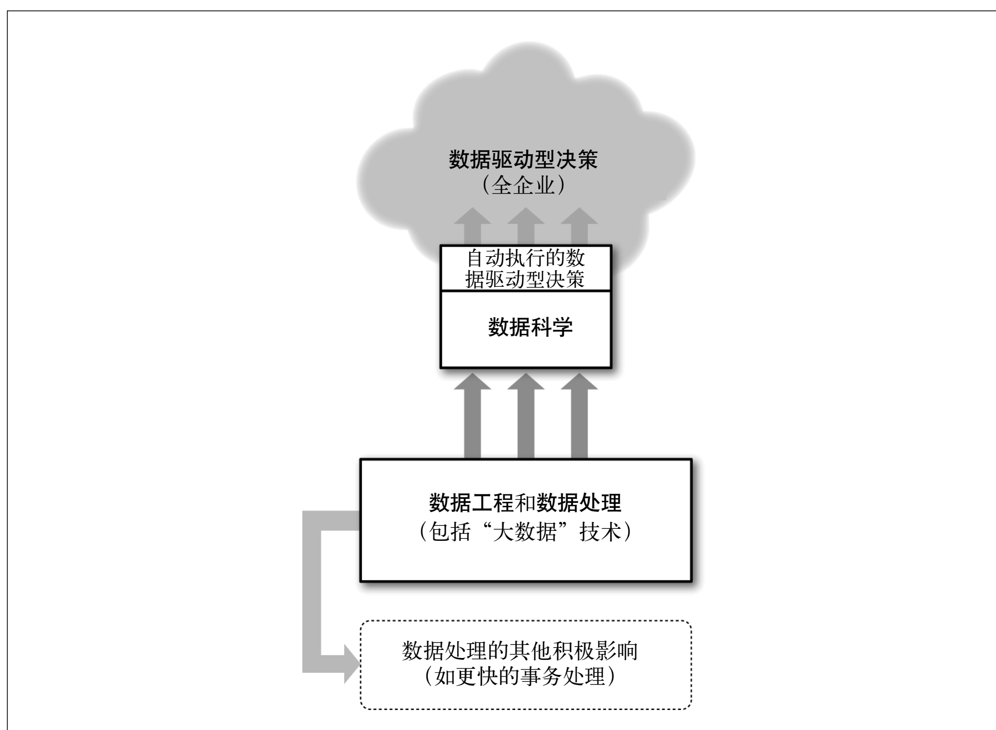


图 1-1：组织中的各个数据处理过程如何运用数据科学

数据驱动型决策（data-driven decision-making，DDD）指的是基于数据分析做出决策，而非仅凭直觉。比如，一位市场营销人员既可以凭多年的从业经验和一双火眼金睛选出最优的一支广告，也可以通过分析顾客对不同广告的反应数据来做决策，还可以把这两种方法结合起来。运用 DDD 不需要在完全依赖它和彻底不用它之间做选择，不同的公司可以不同程度地运用它。

DDD 的优势毋庸置疑。经济学家 Erik Brynjolfsson 及其在麻省理工学院和宾夕法尼亚大学沃顿商学院的同事进行了一项关于 DDD 如何影响公司绩效的研究（Brynjolfsson, Hitt & Kim, 2011）。他们开发出了一种评分方法，用于评估整个公司的 DDD 程度。统计研究表明，公司 DDD 程度越高，其生产力就越高——即使在控制了其他众多可能的混淆因素后，结论也是如此。而且 DDD 的影响不容小觑：得分每增加一个标准差，公司的生产力就相应提高 4%~6%。此外，DDD 不但与资产收益率、股本回报率、资产利用率和公司市值正相关，而且可能与它们存在因果关系。

本书主要关注两类决策：需要从数据中找到“新发现”的决策，以及将会重复做出的决策（特别是大规模重复的决策）。这样一来，即便数据分析仅仅略微地提升了决策的准确度，也能使决策效果得到很大提升。前文提到的沃尔玛案例属于第一类决策：Linda M. Dillman 想发现新知识以帮助沃尔玛做好准备，应对即将来临的飓风 Frances。

2012 年，沃尔玛的竞争对手 Target 百货也因为一次第一类决策而受到了媒体关注（Duhigg, 2012）。和大多数零售商一样，Target 关心顾客的消费习惯、消费动机和影响顾客消费的因素。顾客通常会产生消费惯性，这种惯性很难改变。但是，Target 的决策者们知道，当顾客们的家庭迎来新生儿时，他们的消费习惯就会发生显著变化。Target 的分析师说：“只要能让顾客从我们这里购买尿不湿，他们就会开始从这儿买各种其他商品。”大部分零售商深谙此道，于是他们相互竞争，以期把自己的母婴用品卖给新生儿父母。由于大部分新生儿记录是公开的，因此零售商会基于这些信息针对新生儿父母进行促销。

然而，Target 想在这场竞争中快人一步。他们想预测顾客是否怀孕了，如果预测成功，那么他们就可以赶在竞争对手之前给目标顾客发送母婴用品的促销信息。Target 运用数据科学技术分析了准妈妈们被确认怀孕之前的历史数据，并提取出了能够预测哪位顾客正在怀孕的信息，比如，准妈妈们往往会改变饮食习惯、穿衣风格和维生素摄入方案等。以上种种迹象被从历史数据中提取出来，整合成预测模型，然后应用于市场营销活动。随着内容的深入，本书会详细讨论预测模型。目前你只需要知道，预测模型可以将复杂的世界抽象化、简单化，只关注一系列与我们所关心的问题（比如哪些顾客会流失、哪些顾客会购买、哪些顾客怀孕了等）相关的因素。重要的是，在沃尔玛和 Target 的案例中，数据分析不是为了验证某一假设。相反，分析师探索数据，是为了发现有用的信息。[2]

前文的用户流失案例则属于第二类决策。MegaTelCo 有数亿用户，其中的每一个人都有流失的风险。每个月都有数千万的用户合约到期，因此他们当中的任何一位在近期流失的概率都会不断增加。如果能开发出更加精确的估计方法，可以估计出挽留一位特定用户所带来的收益，那么就可以将其应用到千万级的用户群上，从而收获巨额利润。该思路同样适

注 2：Target 的成功案例也引发了关于使用数据科学技术的伦理问题的讨论。伦理和隐私固然有趣且重要，但是它们目前不在我们的讨论范围之内。

用于其他大量应用数据科学和数据挖掘技术的领域，如直接营销、线上广告、信用评估、金融交易、服务台管理、欺诈检测、检索排名、产品推荐等。

图 1-1 表明，数据科学既支撑着 DDD，也与之部分重合。这指出了一个往往被忽略的事实，即企业越来越多地使用计算机系统进行自动化决策。不同行业使用自动化决策的程度不同。金融业和通信业是较早使用 DDD 的领域，主要原因是它们的数据网络和大规模计算早已成熟，从而实现了大规模的数据聚合和数据建模，以及模型成果在决策中的应用。

20 世纪 90 年代，自动化决策给银行业和消费信贷业带来了巨变，银行和电信公司应用大规模系统来管理以数据驱动的反欺诈决策。随着零售业的信息化程度越来越高，销售决策也越来越自动化。著名的案例有 Harrah’s 赌场的积分项目，以及亚马逊和 Netflix 的自动推荐系统。此时，广告业正经历着一场变革，这主要是因为消费者上网的时间越来越长，以及在线系统瞬间做出广告决策的能力得到了极大提升。

1.5　数据处理和“大数据”

在此有必要谈一下另一点：数据处理过程的许多方面并不属于数据科学。这可能和我们从媒体中得到的印象有些出入。数据工程和数据处理过程都是数据科学中至关重要的支撑，但它们更宽泛。比如，当下很多数据处理技能、系统和技术都被误称为数据科学。要想正确理解数据科学和数据驱动型业务，就必须先理解数据科学与数据工程及数据处理技术的差异。数据科学需要使用数据，它通常得益于基于各种数据处理技术的复杂的数据工程，但这些技术本身并不等同于数据科学。正如图 1-1 所示，这些技术支撑着数据科学，但除此之外，它们的用途还有很多。数据处理技术对于许多面向数据但是与知识获取或 DDD 无关的业务至关重要，例如高效的交易处理、现代 Web 系统处理和线上广告营销管理等。

“大数据”技术（如 Hadoop、HBase 和 MongoDB）最近深受媒体青睐。大数据其实指的是大型数据集，因其过于庞大而无法使用传统的数据处理系统，所以新的处理技术应运而生。和传统技术一样，大数据技术的应用领域也十分广泛，其中包括数据工程。有时，大数据技术也会被用于实现数据挖掘技术。而图 1-1 表明，大名鼎鼎的大数据技术更常用于数据处理，以支撑数据挖掘及其他数据科学行为。

前文提到，Brynjolfsson 的研究展示了 DDD 的优势，而纽约大学斯特恩商学院的经济学家 Prasanna Tambe 进行的另一项研究，则衡量了大数据技术对公司的帮助程度（Tambe, 2012）。在控制了许多可能的混淆因素后，他发现大数据技术的应用程度与显著的额外产出增长相关。具体来说，大数据技术的应用程度每增加一个标准差，公司的生产力就提高 1%~3%；每减少一个标准差，生产力就降低 1%~3%。也就是说，对于两家大数据技术应用程度分别处于两个极端的公司而言，它们的生产力存在天壤之别。

1.6　从大数据1.0到大数据2.0

如果想更好地理解大数据技术的现状，可以类比互联网技术在商业领域的应用过程。在 Web 1.0 时代，各企业想在互联网世界占据一席之地、打造电商业务和提升运营效率，因此忙着采用基本的互联网技术。我们可以认为目前正是大数据 1.0 时代。各个企业正为了

支撑他们目前的运营（如提升效率），而忙着获取大数据处理能力。

一旦完全吸收了 Web 1.0 技术（基础技术的费用也在这个过程中降低了），各个企业就会变得目光长远，开始思考互联网还能做什么，以及如何利用它改进他们的工作。自此，我们便迈进了 Web 2.0 时代：新系统和新公司开始利用互联网的交互性来获益。这种思维转变带来的变化无处不在，最明显的现象就是各种社交网络功能的合并，以及个人客户（和公民）的意见变得越来越难以忽视。

大数据 1.0 时代之后，大数据 2.0 时代指日可待。一旦各个公司能灵活处理大量数据，他们就会想知道："有什么以前做不到的事我们现在能做到了？有什么事现在可以做得比以前好了？"这时很可能就是数据科学的黄金时代。届时，本书介绍的原理和技术可能会得到更深、更广泛的应用。



值得一提的是，一些走在技术前沿的公司在 Web 1.0 时代就早已先于主流应用 Web 2.0 时代的概念了。亚马逊就是极好的例子。该公司早期就注重顾客的意见，并根据这些意见进行产品评级和产品评价（甚至对产品评价进行评级）。同样，可以看到，现在已经有一些企业在应用大数据 2.0 了。比如，亚马逊这回再一次走在了技术的前沿，基于海量数据为其顾客提供数据驱动的商品推荐。还有很多其他的例子。线上广告商不仅需要处理体量极其庞大的数据（每天数十亿的广告曝光量是常事），还得维持极高的货流量（如实时拍卖系统往往几十毫秒之内就会给出结果）。我们应该留意这些行业和其他类似的行业，并从中找出大数据和数据科学进步的迹象，因为这些进步随后必将被应用于其他产业中。

1.7 数据与数据科学能力：一种战略性资产

前几节提出了数据科学的一个基本概念：**从数据中获取有用知识的能力和数据本身，都应被视作关键的战略性资产**。太多企业认为数据分析主要就是从现存数据中发现价值，而往往忽视了企业自身是否有足够的分析能力。而将数据和分析能力都视作战略性资产，就能清醒地认识到该对它们投入多少。我们经常缺乏合适的数据来进行最优决策，或缺乏运用数据进行最优决策的能力，或这两种情况并存。进一步讲，把它们视作战略性资产，还能让我们明白一个事实——它们是**相互补充**的。即使是最优秀的团队，如果没有合适的数据，也难以取得有价值的成果。反过来，如果缺少优秀的数据科学团队，再合适的数据也无法优化决策。和其他资产一样，数据与数据科学能力也需要投资。组建顶尖的数据科学团队虽不寻常，却能给决策带来极大帮助。第 13 章将详细讲述有关数据科学的战略思想。接下来这个案例将说明，对如何投资数据资产有清醒的认识，往往能带来高回报。

小银行 Signet 在 20 世纪 90 年代的经典故事就是一个恰当的例子。早在 20 世纪 80 年代，数据科学就改变了消费信贷业。通过对违约概率建模，这个行业从个人违约风险评估到大规模和市场占有率的战略都发生了变化。这种变化还带来了大规模的伴生经济。虽然现在看起来也许有点奇怪，但当时信用卡的收费标准基本上是统一的，其原因有二：各个企业没有能够处理大规模差异化价格的信息系统；银行管理层认为顾客无法接受价格歧视。

1990 年前后，两位有长远战略眼光的人（Richard Fairbanks 和 Nigel Morris）意识到，信息技术已经足够强大，可以让他们（通过使用本书提及的一些技术）建立更精准的预测模型，并提供差异化服务（比如现今的定价、信贷限额、低原利率余额代偿、现金返还、积分体系等）。可是，这两位没能如愿说服大型银行聘用他们为咨询顾问，也就无从实验他们的想法。在被所有大型银行拒绝后，他们终于得到了美国弗吉尼亚州一家区域性小型银行的青睐，这便是 Signet 银行。这家银行的管理层相信了他们的理论，认为正确的做法是不仅要模拟违约概率，还要模拟收益率。这是因为他们明白，银行的信用卡业务的全部利润其实仅来自于一小部分信用卡用户，而在其他用户身上不是不赚不赔就是亏损。如果他们能模拟收益率，那么就能为最优质的用户提供更优惠的政策，从而把他们从大银行挖走。

然而，Signet 银行在实施这项策略时遇到了大麻烦。他们没有合适的数据来进行收益率建模，也就无法对不同客户进行差异化定价。当时哪家银行都没有这样的数据。由于各家银行一直根据一套特定条款和一个特定的违约模型来发放信用贷款，因此他们只有能模拟他们曾经提供过的条款和他们曾经发放过贷款的客户（即在现有模型中信誉优良的客户）的收益率的数据。

而 Signet 银行能怎么办呢？他们遵循了数据科学的基础策略，即不惜代价地获取数据。一旦把数据看作一项商业资产，我们就要考虑是否投资和投资多少的问题。在 Signet 银行的案例中，银行只能通过实验，给客户提供不同的信贷合约条款，来获得其收益率等数据。随机地给不同客户提供不同条款，这种做法如果脱离数据分析的视角来看会很蠢——你很可能会亏钱！没错，在这个案例中，亏掉的钱恰恰就是收集数据的成本。具有数据分析式思维的人应该关注的问题是，这些数据能否带来符合预期的、与对数据的投资对等的回报。

那么，Signet 银行后来如何了呢？你大概猜到了，因为要收集数据，所以他们随机给客户分配条款，这导致坏账数量暴涨。其坏账率从以前行业领先的 2.9%（即 2.9% 的余额没有被支付）飙升至接近 6%。这样的亏损持续了数年。与此同时，数据科学家们致力于使用这些数据来构建预测模型，评估其效果，最后将其用于提升盈利。因为 Signet 银行把这些亏损当作对数据的投资，所以尽管股东们怨声载道，但是他们坚持了下去。最终，Signet 银行的信用卡业务扭亏为盈并获利丰厚，以至于它最后从银行的业务中拆分了出来。这个成绩至今让整个消费信贷业相形见绌。

Fairbanks 成为了新公司的董事长兼 CEO，而 Morris 成为了总裁兼 COO，他们继续在业务中实践数据科学的概念。这些业务不仅包括用户获取业务，还包括用户留存业务。当一个用户打电话来咨询优惠政策时，以数据驱动的模型会计算各种情形下（采取不同的优惠政策，包括维持现状时）的潜在利润，然后客服代表会向客户报出利润最优的那个优惠政策。

你或许没听说过 Signet 这家小银行，却极有可能听说过那家拆分出来的金融公司：Capital One（第一资本）。Fairbanks 和 Morris 的新公司已经成长为业内最大的一家信用卡发行商，同时它还拥有业内最低的坏账率。据报道，2000 年，这家银行进行了 45 000 项类似的“科学实验”。[3]

注 3：若想了解更多关于 Capital One 的故事，请参考以下资料：Clemons, E. & Thatcher, M.（1998）；McNamee, M.（2001）。

我们很难找到对数据资产的价值进行了清晰量化的研究资料，这主要是因为一般企业不愿意透露有关战略性价值的数据。但是 David Martens 和 Foster Provost 在 2011 年做的一项研究却是个例外，他们对银行用户的交易数据进行评估，衡量了特定数据对银行的优惠政策决策模型的改善程度。银行基于数据建立了这些模型，用来决策向哪些用户推荐哪些产品。此项研究试验了若干类型的数据对预测模型的作用。社会人口学数据可以赋予模型大致区分消费者类型的基础能力，也可以用来预测他们更倾向于购买哪一种产品，但是它也只能做到这些。数据量可以不断增长，但其对模型的贡献却有一个上限。然而，（匿名的）个体消费者的交易明细数据可以大大改进模型。而且这类数据与模型效果的关系清晰且显著：数据量越大，预测模型的表现越好。而且这个趋势在 Martens 和 Provost 的调研范围内没有减退的迹象。这给我们一个重要的启示：拥有较大数据资产的银行跟拥有较小数据资产的竞争者相比，享有重要的战略优势。如果这个趋势可以推广，而且银行有能力进行复杂的分析，那么拥有更大数据资产的银行应该能更好地识别适合每种产品的最优客户，最终结果就是银行产品的采用率增加，或是客户获取成本降低，或两者兼有。

把数据作为战略性资产这一概念既不仅仅适用于 Capital One，也不仅仅适用于银行业。亚马逊很早就开始收集线上用户消费数据，虽然付出了巨额的成本，但是这使用户发现了亚马逊提供的排名和推荐的价值。亚马逊因此能够更容易地留存用户，甚至可以向用户收取一些附加费用（Brynjolfsson & Smith, 2000）[4]。Harrah's 赌场的一项著名投资是收集和挖掘赌场客户的数据，这项投资让它从 20 世纪 90 年代中期的一个小赌场成长为世界上最大的博彩公司（2005 年收购了 Caesar's 娱乐后）。而 Facebook 的巨额估值要归功于其庞大且独特的数据集（Sengupta, 2012），其中包括用户的信息、喜好和社交网络的结构。社交网络的结构对建立预测模型非常重要，因为它可以有效地帮助商家预测什么人会购买特定商品（Hill, Provost & Volinsky, 2006）。当然，虽然 Facebook 拥有非常好的数据，但其是否拥有合适的数据科学策略来发挥这些数据的作用就不得而知了。

随着对数据挖掘原理和数据分析式思维的探索，本书会进一步讨论这些成功故事背后的基本概念。

1.8 数据分析式思维

分析和研究诸如用户流失这类问题，有助于提高“数据分析式”的问题处理能力，而本书的首要目标就是提倡采用这种看待问题的视角。当面对一个商业问题时，你应该能够评估数据是否可以改善这个问题以及如何改善这个问题。我们将探讨一系列基本概念和原理，来促进谨慎思考。同时我们也将开发出一套框架，以便于系统性地组织分析。

前文提到过，即使你从未打算亲自应用数据科学，鉴于数据科学如今在商业策略中的关键地位，理解它也是至关重要的。各个行业越来越多地受数据分析驱动，在这种情况下，有效地与这些行业进行互动或在这些行业中进行互动的能力，将赋予你相当大的专业优势。理解基本概念和掌握用于构建数据分析式思维的框架，不仅可以提升你的互动能力，还可以帮你预见改善数据驱动型决策的机会，以及洞察数据方面的竞争威胁。

注 4：亚马逊推出了付费会员服务。——译者注

许多传统行业的公司正在通过开发新的数据资源或者利用现存的数据资源来获得竞争优势。他们组建了数据科学团队，利用先进的技术来增加收入和降低成本。另外，很多新兴企业正把数据挖掘技术当作关键战略要素来发展，比如 Facebook、Twitter 和其他“Digital 100”企业（Business Insider, 2012）均是由于其业务所获取或创造的数据资产而获得了高额估值。[5] 管理者逐渐开始监督数据分析团队和数据分析项目，市场人员慢慢开始理解和组织数据驱动的市场活动，风险投资者必须明智地投资那些拥有大量数据资产的企业，而企业策划人员必须有能力在方案中利用数据。

再举几个例子：如果一位咨询师给出的提案是通过对数据资产进行挖掘来改善经营状况，那么你应该有能力评定该提案是否行得通；如果你的一个竞争对手宣布他们有了一家新的数据合作方，那么你应该能够判断这是否会使你们在战略上处于劣势。假设你在一家风投公司取得了一个职位，而你的第一个项目就是评估一家广告公司的潜在投资价值。这家公司创始人非常令人信服地提出，他们将通过收集特殊的数据实现巨大的价值，并据此要求提高该公司的估值。这样的要求合理吗？当你理解了数据科学的基本原理时，就应该有能力设计出一连串层层递进的问题，来判断对方关于提高估值的要求是否真的合理。

还有一种规模更小但是更常见的情况，就是各个业务部门都面临着数据分析任务。这些业务部门的员工不得不与数据科学团队打交道。如果他们对数据科学的基本思维方式毫无概念，那么他们恐怕根本就无法理解业务细节。相对于其他技术类项目，这种理解上的缺乏对数据科学项目的破坏性要大得多。由于数据科学是用来支撑更好的决策的，因此数据科学家和业务方面的决策负责人必须紧密合作。下一章会详细讨论这一点。如果一家公司里的业务人员不理解数据科学家的工作，那么这家公司会处于劣势，因为他们会浪费时间和精力，甚至最终可能会做出错误的决策。



管理人员需要掌握数据分析式技能

咨询公司麦肯锡估计：“能让企业从大数据中获益的相关人才短缺。截止到 2018 年，仅美国就短缺 14 万 ~19 万名具有深层分析技能的人才，以及 150 万名能够基于大数据分析结果做出有效决策的管理和分析人才。”（Manyika, 2011）为什么管理和分析人才的缺口是深层分析人才的 10 倍？这当然不是因为数据科学家太难管理，以至于每个科学家需要 10 个管理人员，而是因为，同一业务的不同领域可以使用同一个数据科学团队来辅助决策，提升业务水平。但是正如麦肯锡公司指出的，只有这些不同领域的管理人员理解数据科学的基本原理，才能真正实现业务水平的提升。

1.9 关于本书

本书聚焦于数据科学和数据挖掘的基础知识，囊括了一系列用来搭建数据分析式思维和分析方式的原理、概念以及技术。有了这些基础知识，无须钻研大量具体的数据挖掘算法，就可以深入地理解数据科学的过程与方法。

注 5：当然，这并非新现象。亚马逊和谷歌就是公认的因数据资产而拥有巨大价值的成熟公司。

介绍数据挖掘算法和技术的好书有很多，其中既有实战指南，也有数学书和统计学书。与它们不同，本书只介绍基本概念以及如何使用这些概念来解决数据挖掘的相关问题。但这不意味着可以忽略数据挖掘技术，因为很多算法正是基本概念的具体体现。除了个别几个问题以外，本书不会关注具体技术的细节及它们的运作方式，而是尽可能恰到好处地解释一下细节，以帮助读者理解某项技术的作用以及它所依赖的基本原理。

1.10 重新审视数据挖掘和数据科学

本书花了大量的篇幅介绍如何从大量数据中获取有用的（即重要且最好是可行的）模式或者模型（Fayyad, Piatetsky-Shapiro & Smyth, 1996），以及这种数据挖掘背后的数据科学基本原理。在用户流失预测的案例中，我们可以从之前的用户流失记录中**提取数据**并**获取有用的模式**（如用户行为模式），它既有助于预测将来哪些用户更有可能流失，也有助于设计出更好的用户服务。

本书所介绍的数据科学的基本概念是从很多研究数据分析的领域中总结出来的。尽管对这些概念的介绍将会贯穿本书，但是在此会先做一些简单的描述，以给读者一个大致的感觉。在后续章节会一一详细阐述这些概念。

基本概念：**从数据中获取有用的知识来解决商业问题的过程可以系统地分为若干有明确定义的环节**。“数据挖掘的交叉产业标准”［简称 CRISP-DM（CRISP-DM 项目，2000）］就是这种处理的一个体现。这种处理方式可以提供一个框架，用于组织对数据分析问题的思考。例如，在实践中，尽管我们会反复遇到一些所谓的分析“解决方案”，然而它们却不是基于对问题的谨慎分析或评估得出的。结构化的分析思维则强调那些常常被低估的数据辅助决策的方面，同时这种结构化的思维也有助于更明确地区分人类创造性与高效分析工具的适用范围。

基本概念：**信息技术可以从海量数据中提取出含有信息的、描述目标实体的属性**。用户流失案例中，用户就是目标实体，而每个用户都可以被若干属性所描述，比如用户的使用量、用户使用客户服务的历史记录和许多其他因素。这些属性里面，有哪些会实质性地告诉我们该客户在合约到期时流失的可能性？每个属性又包含多少信息量？回答上述问题的过程有时候被称作“寻找与流失‘相关’的变量”（后续会精确地讨论这个概念）。对此，商业分析师应该做出一些假设并加以验证。他既可以使用分析工具辅助完成这类实验（参照 2.6 节的其他分析技术）也可以（特别是在大规模自动实验的情况下）应用信息技术自动发现含有信息的属性。而且，在根据多个属性来预测流失时，可以递归地应用本概念，后文会对此进行介绍。

基本概念：**如果你过度关注一组数据，那么你或许可以从中获取一些模式，但这些模式可能无法推广至其他数据**。这被称作对数据集的**过拟合**。数据挖掘技术的能力非常强大，因而当它被应用于实际问题时，我们需要识别和避免过拟合。这是我们需要掌握的最重要的概念之一。过拟合，以及避免过拟合的概念将贯穿整个数据科学的过程、算法、评估方法等方面。

基本概念：**阐述和评估数据挖掘的结论时，需要谨慎地考虑它的使用场景**。如果目标是获取可能有用的知识，那么又该如何定义“有用”？这个问题的答案很大程度上取决于它的

应用场景。以用户流失管理的案例为例，究竟应该如何使用从历史数据中获取的模式？除了用户流失概率外，是否还应该考虑用户价值？概括来说，这个模式是否比其他合理的模式更有助于进行辅助决策？如果不使用任何模式，随机决策，效果会如何？如果使用一个智能的预设状况来替代，效果又如何呢？

以上四条仅是将要探讨的数据科学基本概念中的一部分。本书将详细讨论十几条这样的基本概念，并大体演示它们如何帮助我们构建数据分析式思维以及理解数据挖掘技术、算法和数据科学的应用。

1.11 数据科学：一门新兴的实验性学科

在继续之前，应该简要回顾一下数据科学的工程应用。撰写本书之际，人们谈论数据科学时，不仅会谈到用于解读数据的数据分析技能和技术，还会提到常用的数据科学工具。数据科学家的定义（以及招聘广告中的职位描述）中不仅会明确专业领域，还会明确具体的编程语言及工具。招聘数据科学家的广告中经常会提及数据挖掘技术（如随机森林、支持向量机）、具体的应用领域（如推荐系统、广告布局优化）以及常用的大数据处理软件（如 Hadoop、MongoDB）。通常，人们很少明确区分数据科学和大型数据集处理技术。

必须指出，数据科学和计算机科学一样，是一个年轻的领域。大众刚刚开始特别地关注数据科学，而其基本原理也刚开始出现。数据科学如今的状态可以类比 19 世纪中叶的化学科学，那时候化学理论和化学基本原理日渐规范化，而这个领域又是非常依赖实验的，因此当时每位优秀的化学家都必须是一位合格的实验室技术员。与之相似，现在一名合格的数据科学家也必须能够熟练使用特定的软件和工具。

总而言之，本书聚焦于科学而非技术。这里没有关于在 Hadoop 集群上执行大数据挖掘的最佳实践指导，甚至没有 Hadoop 的定义或学习它的理由。[6] 本书聚焦于数据科学中业已形成的基本原理。10 年后，占主导地位的技术很可能会改变或进步，而我们现在对技术的讨论也会过时，但是，鉴于基本原理现在仍与 20 年前相同，所以它们在接下来的 10 年中极有可能仍然变化甚微。

1.12 小结

本书的主题是如何从大数据中获取有用的信息和知识，以改善商业决策。当今，几乎所有行业部门和业务单位都积累了海量的数据，而数据挖掘的机遇也已经遍布各行各业。潜藏在数据挖掘技术庞大身躯下的，是一套更加简洁的基本概念，而这套基本概念构成了数据科学。这些概念是普适的，囊括了数据挖掘和商业分析的大部分精髓。

若想在当今数据导向的商业环境中取得成功，就必须考虑如何将数据科学的基本概念应用到具体的商业问题上，也就是要进行数据分析式的思考。例如，本章提到过，数据应该被视为一项商业资产。一旦确立了这个思考方向，我们就会开始考虑投资于数据的必要性（和力度）。因此，理解数据科学基本概念，不仅对数据科学家本身至关重要，对任何与数

注 6：Hadoop 是一个应用广泛的、高度可并行的开源计算框架，是当今用于处理超过常规数据库系统处理能力的大型数据集的“大数据”技术之一。Hadoop 是基于谷歌提出的并行处理框架 MapReduce 开发的。

据科学家共事的人、聘用数据科学家的人、投资重数据资产的人，以及各机构中领导数据分析应用的人同样至关重要。

构建数据分析式思维离不开概念性框架的帮助（本书会通篇讨论后者）。例如，下一章的主题——从数据中自动提取模式——就是一个可分为明确环节的流程。理解这些流程和环节有助于构建数据分析式思维，使之更加系统化，并减少错误与遗漏。

事实证明，数据驱动型决策和大数据技术可以显著提升经营业绩。数据科学支撑着（有时也执行）数据驱动型决策，同时依赖于“大数据”存储和工程技术，但是数据科学的原理是独立的。本书所讨论的数据科学原理与其他重要的技术（如统计假设检验和数据库查询，读者可另寻相关图书和课程学习）既相互区别，又相互补充。下一章将详细探讨它们的区别。

第 2 章

商业问题及其数据科学解决方案

基本概念：一系列典型数据挖掘任务；数据挖掘流程；有监督型数据挖掘与无监督型数据挖掘

数据科学的一条重要原则是，数据挖掘的**流程**可以分解为几个通俗易懂的环节。有些环节涉及信息技术的应用，如数据中模式的自动发现和评估，而有些则主要依赖数据分析师的创意、常识和商业知识。理解数据挖掘的整个过程，有助于组织数据挖掘项目，使它们更接近系统性的分析，而不是凭借运气和个人智慧的冒险行为。

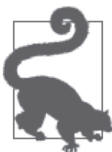
数据挖掘流程把从数据中找出模式这一任务分解成了一系列定义明确的子任务。这种方式对组织对数据科学的讨论也很有用。本书将会把该过程作为讨论的主要框架。本章将介绍数据挖掘的整个过程。但是在此之前，需要先讲一下各类常见的数据挖掘任务，这样，在接触数据挖掘的整个流程和后续章节中的其他概念时，本书会更加言之有物。

本章最后会讨论一系列商业分析主题，如数据库、数据仓库和统计学基础。尽管这些主题不是本书的重点，但它们也非常重要。读者可以参考其他图书（这样的书有很多）来学习这些主题。

2.1 从商业问题到数据挖掘任务

每个数据驱动的商业决策问题都是独一无二的，因为其包含的目标、愿望、约束，乃至问题中的人物个性都不尽相同。但和许多工程问题一样，归根结底，商业问题也可以被分解为许许多多的普通任务。与企业利益相关方合作时，数据科学家往往会把一个具体的商业问题分解成一个个子任务。将子任务逐一解决，再将其解决方案组合起来，就构成了整个问题的解决方案。这些子任务中，有的是该商业问题中所特有的，而其他的都是常见的数据挖掘任务。比如 MegaTelCo 公司的电信用户流失问题就是该公司特有的，因为其中的

某些细节必然和其他电信公司的用户流失问题不同。然而，基于历史数据预测用户在合约到期后不再续约的概率，这一子任务很可能是所有用户流失问题的解决方案的一部分。如果把 MegaTelCo 的具体数据转化成特定格式（这个问题将在下一章阐述），那么这个用户流失概率的估计问题就会转化成一个非常常见的数据挖掘任务，而无论在理论方面还是实践方面，我们都非常了解如何解决常见的数据挖掘任务。后面的章节还会提供数据科学框架，以便于将商业问题分解为子任务，以及将子任务的解决方案重新组合。



数据科学中一项至关重要的技能，就是把一个数据分析问题分解为若干有现成工具可用的已知任务。识别出旧问题及其解决方案，不仅有助于避免重复劳动，节约时间和资源，还能让我们专注于问题中更有趣的部分：那些尚未自动化的、仍旧依赖人类的智慧和创意进行解决的部分。

尽管多年来，大量具体的数据挖掘算法被开发了出来，然而归根结底，它们都用于解决几类基础任务。因此，有必要明确定义这几类基础任务。接下来的几章里，本书会先用两类任务（分类和回归）来阐明几个基本概念。之后，本书将使用“个体”一词指代数据中的实体，如用户或消费者这样的自然人，或者公司这样的无生命实体。第 3 章将对这个术语进行更精准的解释。在许多商业分析项目中，我们往往想找出描述个体的变量与其他变量之间的相关关系，比如，虽然从历史数据中可以知道哪些用户在合约到期后离开了公司，但我们更想找出哪些变量与用户是否会在近期流失真正相关。而寻找这种相关关系正是分类任务和回归任务的最典型例子。

(1) **分类**和类**概率估计**可以用于估计总体中的每个个体在一（小）组类别里到底属于哪一类。通常这些类都是排他的。举个分类问题的例子：在“MegaTelCo 的所有用户中，哪些人可能对促销活动做出响应？”那么这组类别里就有两个类别，即“会响应”和“不会响应”。

   在分类任务中，数据挖掘过程会产生一个模型，而这个模型能决定给定个体被归入哪一类。与分类密切相关的任务被称为**评分**或类**概率估计**。评分模型在应用于个体时，不会预测类别，而会输出表示该个体属于各类的概率的评分（或其他量化可能性的指标）。在前文的例子中，评分模型能够对每个用户进行评估，并输出他们响应促销活动的概率。分类与评分密切相关，以后我们会看到，这两种任务实际上可以相互转化。

(2) **回归**（“值估计”）可以用于估计或预测每个个体的某个变量的数值，例如：“某顾客对这项服务的使用量是多少？”此例中需要预测的变量是**服务使用量**。我们可以利用总体中的其他相似个体及其历史数据来构建预测模型，而回归程序就能输出用于估计给定个体的特定变量的值的模型。

   回归与分类既相互联系，又相互区别。通俗地说，分类是在预测某事**是否**会发生，而回归则是在预测某事**有多大可能**发生。这种区别将随着学习的深入逐渐明晰。

(3) **相似性匹配**可以基于已知数据识别出相似的个体。它可以直接用于找出相似的实体。例如，IBM 想找出与其最佳客户相似的企业，以便将销售资源尽可能多地配置在它们身上。于是他们基于“企业造影”数据——描述企业特点的数据——来进行相似性匹配。相似性匹配是一种常用的商品购买推荐（依据人们在产品方面的喜好或购买记录，来寻

找与你相似的人）方法的实现基础。度量相似性也是解决其他数据挖掘任务的基础，如分类、回归和聚类。第 6 章将详细讲解相似性及其用途。

(4) **聚类**可以用于在没有具体目标的情况下，根据相似性将个体**归为若干组**。例如："客户是聚集成自然组群还是被划分成了不同部分？"聚类在初步的领域探索中非常有用，它可以找出可能存在的自然组群，而这些群组会给下一步的数据挖掘任务和方法提供线索。聚类还能作为信息输入到某些决策过程中，以帮助回答"应该提供或开发哪些产品""客户服务团队（或销售团队）应如何构建"等问题。第 6 章将深入探讨聚类。

(5) **共现分组**（又名频繁项集挖掘、关联规则发现和购物篮分析）可以用于根据交易记录找出个体之间的关联。例如：**顾客往往会同时购买哪些商品**？聚类方法根据对象的属性寻找对象间的相似性，而共现分组则根据对象是否在交易中同时出现来判断其相似性。比如，在分析超市的销售记录后，你可能会发现人们同时购买碎肉和辣椒酱的频率高得出人意料。尽管利用这样的发现也许需要一些创造力，但是一般情况下，可以据此推出促销活动、改变商品摆放方式或进行搭配销售。把常常同时卖出的商品分在同一组，是一种常见的分组方式，它也被称为"购物篮分析"。某些**推荐**系统也会通过找出哪两本书常常被人一起购买（"买了 X 的人往往也买 Y"）来进行类同分组。

共现分组的结果是对共同出现的个体的描述，其中包括对它们共现频率的统计，以及这种频率是否有显著意义。[1]

(6) **画像分析**（又名行为描述）可以用于描绘个体、群组或总体的典型行为特征。例如："被划分出的某组用户典型的手机使用量是多少？"描述行为并不是一件简单的事情。要对手机使用量进行画像分析，就需要分别对夜间和周末的平均通话时长、国际通话使用数据、漫游收费数据、短信使用数据等进行复杂的描述。我们既可以对整个总体的行为进行泛泛的描述，也可以具体地对小型群组甚至个体进行分析。

画像分析常用于为异常检测建立行为标准。其具体应用有欺诈检测和计算机系统入侵监控（比如当你的 iTunes 账户被黑的时候）。举个例子，如果知道某用户平时用信用卡消费的习惯，就能判断卡上某笔新消费是否符合该用户的画像。我们可以把"错配程度"转化为"可疑得分"，若分数太高，就要向顾客发出警告。

(7) **链路预测**可以用于预测数据项之间的联系，其方法通常是，假定某链路存在并估计该链路的强度。链路预测在社交网络中非常常用，例如："您和 Karen 有 10 名共同好友，您是否愿意把 Karen 加为好友？"链路预测还能用于估计链路的强度，比如，在向用户推荐电影时，可以构造一张链路图来描述用户和他们看过或评价过的电影之间的联系。从这张图中，可以找出用户和电影之间那些并不存在，但经过预测应该存在且强度很大的链路，而这些链路就构成了电影推荐的基础。

(8) **数据整理**是将大数据集转化为保留了重要信息的较小数据集的过程。小数据集处理起来更简便，而且从中获取信息可能更为容易。比如，一个庞大的消费者观影偏好数据集可以被整理成较小的、能体现数据中隐含的消费者偏好（如观影者对电影题材的偏好）的数据集。虽然数据整理通常会造成部分信息的流失，但重要的是它提升了我们对数据的洞察。

注 1：某些共现个体组合并不出人意料，比如瓶装水可能总和其他商品同时出现在购物篮里，因此它跟某种其他商品的共现组就没有显著意义。——译者注

(9) **因果模型**能帮助我们理解哪些事件或行为对其他事件产成了实质性的影响。例如，假设我们用预测模型进行精准广告投放后，发现目标用户在广告触达后的购买率的确比之前更高，但这是因为广告确实影响了用户的行为，还是因为预测模型选择了本来就有购买计划的那些用户？因果模型的技术涉及对数据的大量投入——如随机对照试验（比如所谓的“A/B 测试”）——和根据观测数据得出因果结论的复杂方法。因果模型的实验方法和观测方法通常可被视为“反事实”分析：它们研究在互斥条件下，目标事件（如精准广告触达特定个体）发生和不发生两种情形之间的区别。

无论何时，细心的数据科学家都应在因果模型的结论中明确该结论所依据的假设（这样的假设**一定**存在）。在进行因果模型分析时，需要根据商务需求来权衡是增加对数据投入以减少对假设的依赖，还是在当前假设下接受分析结论。即使是在最严谨的随机对照试验中，为了防止因果分析结论无效，也必须做假设。众所周知，医药研究中的“安慰剂效应”之所以被发现，就是因为在精心设计的医药随机对照试验中，研究人员忽略某些假设条件而导致试验结论无效。[2]

如果要详细讨论以上所有的数据挖掘任务，那么就要写好几本书了。因此，本书只展示数据科学最基础的一些原理，而这些原理共同构成了上述所有任务的基础。本书将主要使用分类、回归、相似匹配和聚类等任务来阐明这些原理，在需要时也会讨论其他有助于理解基础原理的任务（直至本书结尾）。

请思考一下：解决用户流失预测问题需要使用以上任务中的哪几种？在实践中，流失预测通常会被转化为通过划分用户，找出哪部分用户更可能离开公司的问题。这个问题似乎可以使用分类任务、聚类任务甚至回归任务来解决。然而哪一个才是最佳选择呢？要回答这个问题，首先需要了解一下它们的区别。

2.2 有监督方法与无监督方法

请思考下面两个相似的、有关用户群的问题。第一个问题是：“用户是否能自然地分成不同群组？”这个分组任务并没有任何明确的目标或目的，而这种没有**目标**的数据挖掘问题就被称为**无监督**的数据挖掘问题。另一个非常相似的问题是：“能否找到在合约到期后极有可能不续约的那群用户？”此处出现了特定目标：客户在合约到期后会不会续约？在此问题中，我们是出于“基于流失概率而采取行动”这一原因进行分类，这被称为**有监督**的数据挖掘问题。



术语解释：有监督学习和无监督学习

“有监督”和“无监督”这两个术语来源于机器学习领域。打个比方：在有监督学习的情况下，老师通过提供目标信息和一系列示例来“监督”学员学习；无监督学习尽管可能会涉及相同的示例集，但不会有人提供目标信息，学员不知道学习目标，因而需要自己通过总结示例的共同特征得出结论。

注 2：“安慰剂效应”指病人虽未获得有效治疗，却因“相信”治疗有效，而让症状得到舒缓的现象。
——译者注

以上两个问题的差别虽然细微却极其重要。如果给出了确定的目标，那就是有监督的问题。有监督型数据挖掘任务所需的技术不同于无监督型数据挖掘任务，但其结论往往更为有用。有监督型数据挖掘任务会给定一个分类目标，即预测目标的类别。而像聚类这种无监督的任务则根据相似性对个体进行分组，然而它无法保证这种相似性有意义或能用于任何具体目的。

有监督型数据挖掘在技术上还要求一个必要条件：必须有目标**数据**。这里的目标不能仅在理论上存在，还必须实实在在地存在于数据中。例如，你可能需要知道某个特定用户是否会在至少 6 个月内继续留存，但如果这种留存信息在历史数据中缺失或不完整（比如只有两个月的留存数据），你就无法达到目的。获取目标数据往往是数据科学投资的重点。个体目标变量的值通常被称作个体的**标签**，意在强调在标注数据时往往（并非总是）会产生费用。

分类模型、回归模型和因果模型通常用有监督方法构建；相似匹配、链路预测和数据整理采用两种方法皆可；聚类、共现分组和画像分析则通常用无监督方法解决。这些分析方法的基础就是我们要展开讨论的数据科学的基本原理。

回归与分类是两类有监督型数据挖掘方法，两者的区别在于目标变量的类型不同。回归的目标变量是数值型，而分类的目标变量则是类别型（通常是二元型，即 0-1 类型）。下面几个问题很相似，它们均需要采用有监督型数据挖掘方法来处理。

> “得到激励 I 的顾客会购买服务 S1 吗？”
>
> 这是一个分类问题，因为其目标变量是二元的（顾客买或不买）。
>
> “得到激励 I 的顾客会购买哪种服务组合（S1、S2 还是都不买）？”
>
> 这是一个含有三元目标变量的分类问题。
>
> “该顾客使用该项服务的程度有多大？”
>
> 这是一个回归问题，因为其目标变量是数值型，即每位顾客的服务使用量（实际值或预测值）。

上述问题有几处细节需要注意。在实际的商业应用中，我们往往更想得到预测数值而非类别。例如，在用户流失示例中，关于用户是否会继续订购服务的结论可能并不足以满足需求，我们想要的是用户续约的**概率**。但这仍是一个分类问题而非回归问题，因为其中的目标变量是类别型。为了避免混淆，我们称之为“类概率估计”。

在数据挖掘流程的初始环节，重点是判断首要的分析方法是有监督的还是无监督的。如果是有监督的，那就需要给予目标变量精准的定义。该目标变量必须是具体的量，它会成为数据挖掘的焦点，其取值可以从示例数据中获得。第 3 章将再度讨论这个问题。

2.3　数据挖掘及其结果

在数据挖掘中，发现模式并建立模型与**使用**数据挖掘结果之间的区别也很重要。在学习数据科学时，学生常常会混淆两者；在讨论数据分析时，管理人员有时也会分不清两者。如

何使用数据挖掘结果的确会影响和指导数据挖掘流程本身，但需要注意：这两者是截然不同的。

请回想用户流失示例的应用场景。我们想用模型预测哪些用户会流失。假设已经通过挖掘数据建立了类概率的预测模型 M，那么输入某个现有用户的一系列属性后，模型 M 将输出该用户流失的分数或概率。这就是数据挖掘结果的**使用**，而数据挖掘往往是通过其他历史数据得到模型 M 的。

图 2-1 展现了以上两个环节。数据挖掘产生概率估计模型（如上半幅图所示），该模型随即被应用到另一个未知的示例上，并输出估计概率（即模型使用环节，如下半幅图所示）。

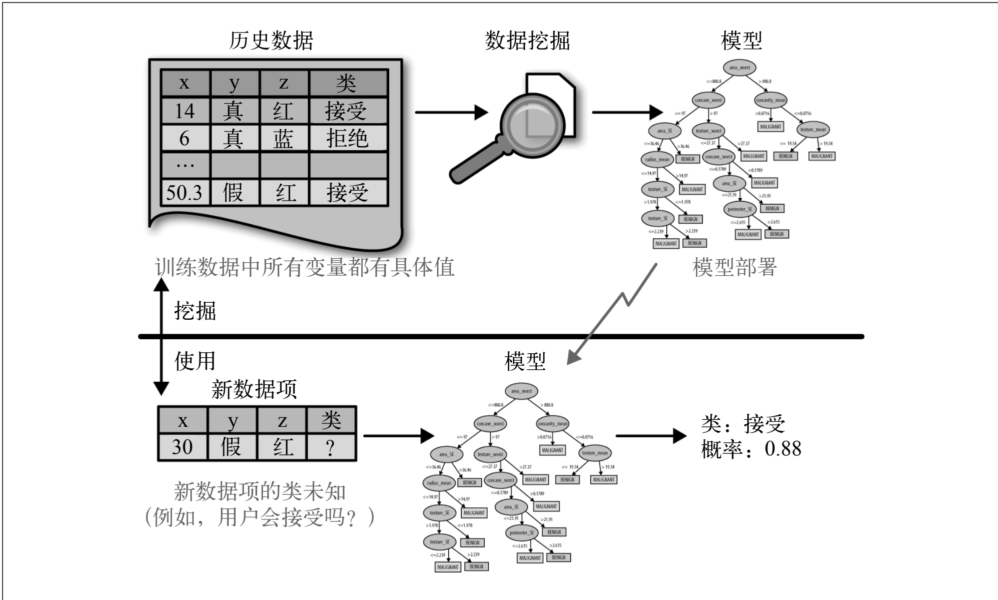


图 2-1：挖掘数据与使用数据挖掘结果的区别。上半幅图展现的是通过挖掘历史数据得到模型的过程。注意，历史数据是明确标注了目标值（类别）的。下半幅图展现的是数据挖掘结果的应用，即把模型应用于类别未知的新数据。模型最终预测了类别以及该类别的概率

2.4 数据挖掘流程

数据挖掘是一门手艺。它涉及大量科学与技术的应用，而如何恰当地应用这些科学与技术也是一门艺术。但如同其他成熟的手艺一样，数据挖掘也有一套易于理解的流程，可以将问题解构，并保证合理的一致性、可重复性和客观性。跨行业数据挖掘标准流程（CRISP-DM; Shearer, 2000）对该流程进行了整理，如图 2-2 所示。[3]

注 3：你也可以访问 CRISP-DM 流程模型的维基百科页面。

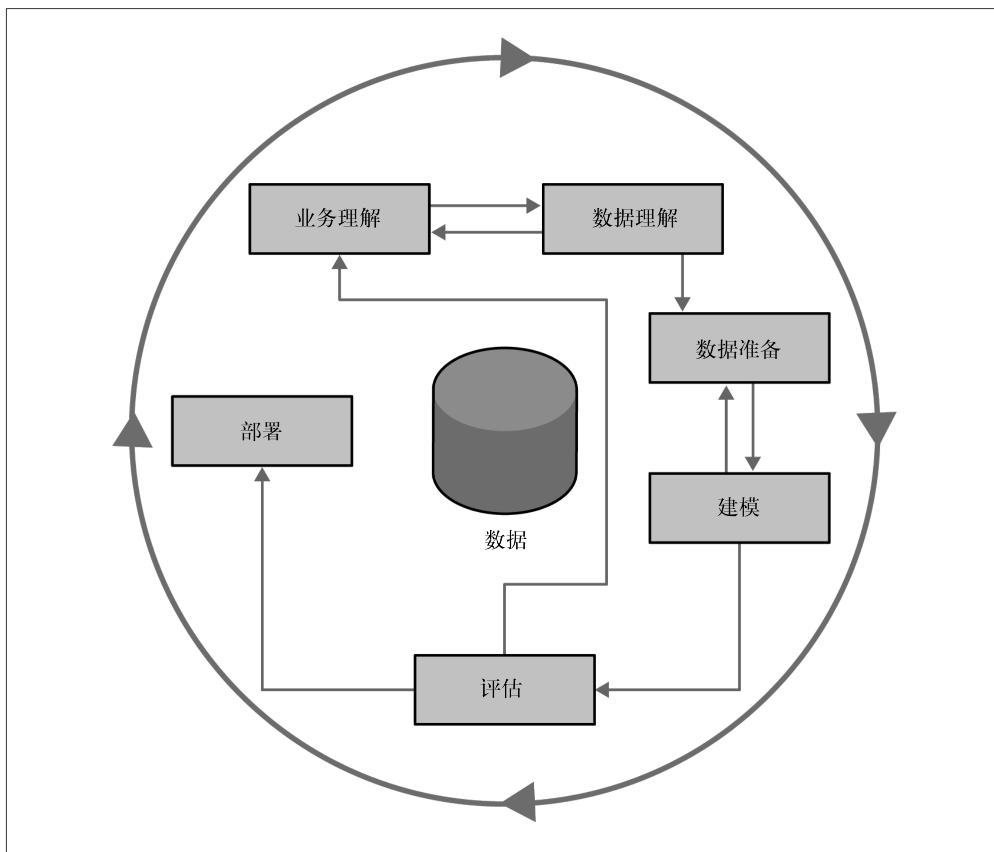


图 2-2：CRISP-DM 数据挖掘流程

图 2-2 明确了这个事实：循环迭代是数据挖掘流程的常态。通常，循环迭代一轮没能解决问题并不意味着失败。整个数据挖掘流程往往是探索数据的过程。在第一轮迭代之后，数据科学团队会对数据更加了解，在下次迭代时会更有方向性。下面详细讨论这些步骤。

2.4.1　业务理解环节

在初始环节，理解要解决的问题至关重要。虽然这似乎显而易见，但是实际上商业项目很少会像数据挖掘问题那样清晰明确。因此，在探寻结果的过程中，需要反复重塑问题和设计解决方案。如图 2-2 所示，该过程不是一个简单的线性过程，而是一个嵌套在循环中的循环。因为最初的构想可能是不完整的或不是最优的，所以若要得到满意的解决方案，就需要进行多次循环。

在业务理解环节，分析师需要发挥创造力。正如接下来将会讲到的，数据科学固然重要，但成功的关键往往是商业分析师如何发挥创造力，将商业问题分解成一个或多个数据科学问题。充分掌握基础知识有助于分析师构思新颖的方案。

在解决特定的数据挖掘问题时，有多种工具可供使用，详见 2.1 节。通常在前期我们会依据这些工具各自的优势来设计解决方案，也就是说，把问题分解为若干个分类任务、回归任务和概率估计任务的建模问题。

在第一个环节，**方案设计团队需要仔细考虑需要解决的问题及其应用场景**。这是数据科学最重要的基本原则之一，本书将用整整两章（第 7 章和第 11 章）来讨论。我们究竟想做什么？如何做？应用场景的哪些部分可能包含数据挖掘模型？在详细讨论上述问题时，本书会先采用一个简化的应用场景。但随着学习的深入，本书会折回来，根据实际业务需求不断调整应用场景。本书还会介绍一些概念性工具来辅助思考，比如，根据期望值来构建商业问题，有助于把问题系统地分解为多个数据挖掘任务。

2.4.2 数据理解环节

如果目标是解决商业问题，那么构成解决方案的原材料就应该包含在数据里。因为几乎没有一份数据能完全符合问题的需求，所以我们需要了解数据的优势和局限性。收集历史数据的原始目的往往与当前面对的商业问题无关，有些历史数据甚至根本没有明确的收集目的。另外，用户数据、交易数据和市场回馈数据包含不同的信息，其涵盖的交叉人群和数据的可靠程度也可能不同。

数据的**成本**不一也是常见现象。有的数据几乎可以免费获取，有的却需要费点力气才能获得。有的数据可以买到，有的数据却根本不存在，而采集它们甚至需要一个辅助项目。数据理解环节的关键是估计每个数据源的成本和收益，从而决定是否有必要进一步投资。即使所有数据集都收集齐全时，也需要额外花力气对其进行核对。比如，众所周知，用户记录和产品识别码多变且杂乱，清洗数据并匹配用户记录，以确保用户和记录一一对应，这本身就是一个复杂的分析问题（Hernández & Stolfo, 1995; Elmagarmid, Ipeirotis & Verykios, 2007）。

随着数据理解环节的深入，问题解决路径的方向可能会随之改变，而数据团队的工作甚至有可能产生分支。以欺诈检测为例。数据挖掘技术被广泛地应用于欺诈检测，而许多欺诈检测问题都涉及经典的有监督型数据挖掘工作。请思考一个信用卡反欺诈示例。因为消费记录会呈现在每个用户的账户里，所以盗刷行为很容易被发现——要么一开始被信用卡公司发现，要么事后在查看账户记录时被顾客发现。因为合法用户和欺诈罪犯是目的完全相反的、截然不同的人，所以可以假设几乎所有的欺诈行为都被识别并合理标注了。因此信用卡交易就有了可靠的、能作为有监督型数据挖掘的目标变量的标签（**欺诈**和**合法**）。

现在请思考另一个相关问题：反医保欺诈。这个问题在美国每年都会耗费数十亿美元。尽管它看上去很像一个传统的欺诈检测问题，但只要思考一下这个问题与数据的关系，就能意识到，这两个问题其实迥然不同。这个问题中的欺诈者——提出虚假保险赔付申请的医疗机构或患者——也是医保系统中的合法医疗机构和合法用户。因为欺诈者是合法用户的一部分，所以并不存在一个客观公正的中立方来告诉我们“正确”的收费价格应是多少。其结果就是，医保费用数据没有可靠的、能标注欺诈行为的目标变量，因此，适用于信用卡欺诈的有监督学习方法也就不适用了。这样的问题通常需要用无监督方法，如画像分析、聚类、异常检测和共现分组来解决。

以上两个问题似乎都是欺诈检测问题，然而这种相似仅仅是表面上的，而且非常具有误导

性。在数据理解环节，需要深挖到表面之下，来发掘商业问题的结构和可用的数据，然后把它们和一种或多种数据挖掘任务相对应，以充分应用科学和技术手段来解决问题。通常，一个商业问题会对应多个数据挖掘任务，而且这些任务往往种类不同，因而有必要将其解决方案进行组合（详见第 11 章）。

2.4.3 数据准备环节

虽然我们可以运用的分析技术十分强大，但是它们对所使用的数据有一些特定要求。通常，因为它们要求的数据格式与数据产生时的原始格式不同，所以需要对数据进行转化。因此，数据准备环节往往紧跟着数据理解环节，而在此环节中，数据被处理转化成特定格式，以获得更好的结果。

典型的数据准备的例子有：把数据转化为表格格式、删除或推断出缺失值，以及转换数据类型。有的数据挖掘方法适用于符号数据和分类数据，有的则适用于数值型数据。此外，数值往往需要归一化或调整比例，以便于进行比较。上述几种转换都有相应的标准技术和经验法则。第 3 章将详细讨论数据挖掘所用的几种最典型的数据格式。

然而，本书不会重点讨论数据准备技术，因为该话题本身即可单独成书（Pyle, 1999）。接下来的几章将只定义一些基本的数据格式，而且仅在需要其帮助理解数据科学的基本原理或更好地展示具体示例时，才会详细解释数据准备。



通常来说，数据科学家往往会在初期投入大量时间来定义之后会用到的变量，而这是人的创造力、常识和商业知识发挥作用的主要时期。通常，数据挖掘结果的好坏，主要依赖于分析师能否非常好地构建问题和设计变量（虽然有时他们很难承认这一点）。

在数据准备环节，有一个非常常见且重要的问题需要注意，这就是"漏洞"（Kaufman 等 , 2012）。漏洞是指，虽然历史数据中的一个变量会提供有关目标变量的信息，但这些信息在需要进行决策时尚不存在。举个例子。"一个网络会话中的网页访问总数"这个变量可以用来预测某个上网者在某个特定时间点是会结束浏览该网站，还是会继续访问网站的其他页面。但该变量的值在该会话结束前是未知的；而在会话结束后，就可以直接得到目标值，而不需要通过预测来推断它了（Kohavi 等 , 2000）。再举一个例子。对于预测顾客是否是一名"土豪"而言，其所购商品的种类（或退而求其次，支付的税额）非常有用，可是在预测时是不可能知道该变量的值的（Kohavi & Parekh, 2003）。在数据准备环节，必须慎重考虑漏洞问题，因为数据准备往往是基于已发生的事实（即历史数据）的。第 14 章会更详细地展示一个难以发现的漏洞问题的真实示例。

2.4.4 建模环节

由于建模环节是接下来几章的主题，所以在此处不加赘述。但不得不说的是，建模环节所输出的就是能反映数据中的规律的模型或模式。

建模环节是将数据挖掘技术应用于数据的主要阶段。你需要理解数据挖掘的基本概念，包括现有技术和算法的种类，因为科学和技术正是在数据挖掘的这个环节发挥了最大的作用。

2.4.5 评估环节

评估环节的目的是严格评估数据挖掘结果，以确保它们有效且可靠，能够用于下一步。只要仔细地探查一个数据集，总能从中发现各种模式。但在严格的审查下，这些模式却可能根本站不住脚。我们希望能确保从数据中提取出的模型和模式能体现真正的规律，而不是特殊情况或样本异常。你可以在数据挖掘结束后直接应用其结果，但我们不建议你这么做。通常，在应用某个模型之前，先在可控的实验室环境下对其进行测试，才是更加简单、廉价、快速和安全的做法。

同样重要的是，评估环节还能确保模型满足最初的商业目的。要记得，商业中数据科学的首要目标是辅助决策，而且从开始应用数据挖掘起，我们就聚焦于想要解决的商业问题。通常，数据挖掘给出的解决方案只是一个大方案的一部分，它同样需要评估。即使模型在“实验室”里经受住了严格的评估，也可能会因某些其他外部因素而失效。比如，检测解决方案（如欺诈检测、垃圾邮件检测和入侵监测）的通病就是误报太多。同一个模型，在实验室标准下可能极其准确（>99%），而在实际商业背景下却可能由于出现过多误报，以致成本太高而无法使用。（想一想：处理所有误报信息的成本是多少？而安抚客户不满情绪的成本又是多少？）

数据挖掘结果的评估环节包含定量评估和定性评估。企业的各种利益相关者都关心数据挖掘输出的最终模型做出或者辅助做出的商业决策。在许多情况下，模型的应用需要得到他们的“同意”，而他们同意的前提是对模型决策的质量感到满意。上述情境会因应用而异，但利益相关者们往往想知道应用这个模型是否利大于弊，尤其是该模型会不会出现致命错误。[4] 为了促成这种定性评估，数据科学家必须要考虑模型对于企业利益相关者（而不仅仅是数据科学家）而言的**可理解性**。如果模型本身就令人费解（比如有许多复杂的数学公式），那数据科学家又怎么能让模型的表现易于理解呢？

最后，具备一个综合评估框架是非常重要的。这是因为，从一个已经部署的模型中获取其表现的详细信息是十分困难的，有时这甚至是不可能的。首先，由于对部署环境的访问受到限制，所以“生产中”的综合评估就变得非常困难。其次，已部署系统通常包含许多“活动部分”，因而对每个单独环节进行评估也很困难。鉴于这种情况，拥有成熟数据科学团队的公司会明智地搭建尽可能反映真实生产数据的实验环境，以便在部署模型前得到最真实的评估。

尽管如此，在某些情况下，我们也想通过构建能进行随机化试验的实时系统等方法，在开发环节进行评估。在用户流失示例中，如确定数据挖掘产出的某个模型能使用户流失情况好转，那么我们下一步可能会进行“活体”评估，即实时系统将模型随机应用于某些用户，而将其他用户作为对照组（还记得第 1 章讨论的因果模型吗）。这样的实验必须经过精心设计，但因其技术细节超出了本书的讨论范围，在此不做讨论。感兴趣的读者不妨读一读 Ron Kohavi 等人的文章（Kohavi 等 , 2007, 2009, 2012）。我们还想对已部署的系统进行评估，以确保外界环境的变化不会对模型决策造成负面影响。比如，有些事件（如欺诈

注 4：比如，在某数据挖掘项目中，为了诊断当地电话网络的故障并向疑似的故障地点派遣技术人员，人们构建了一个模型。在部署该模型之前，电话公司的一些利益相关者要求对模型进行微调，以便对各医院进行特殊处理。

或垃圾邮件）的表现在模型部署后会即刻发生改变。另外，模型的输出很大程度上依赖于其输入数据，而输入数据的格式或者内容经常会在数据科学团队不知情的情况下发生改变。Raeder 等人（2012）为了帮助处理诸如此类的已部署系统评估及相关的问题，对系统设计进行了详细探讨。

2.4.6 部署环节

在部署环节，数据挖掘结果乃至数据挖掘技术本身正（越来越多地）付诸实际使用，以获取投资回报。最简明的一类示例涉及在某些信息系统或业务流程中实现预测模型。在用户流失示例中，可以将预测流失概率的模型接入用户流失管理系统，这样，管理系统就可以向那些极有可能离开公司的用户发送特殊优惠（后文中会进一步探讨）。我们还可以将新型欺诈检测模型嵌入到劳动力管理信息系统中，以监视用户账户并“挑出”可疑交易交予欺诈分析师进行人工检验。

数据挖掘技术本身越来越多地被直接部署。比如，在精准投放线上广告时，我们会选择部署能在新广告宣传活动出现时，能自动批量构建（并测试）模型的系统。之所以部署数据挖掘系统而非其产出的模型，主要原因有二：一是诸如欺诈检测和入侵监测一类的技术变化太快，数据科学团队难以招架；二是一个企业要构建的商业模型太多，数据科学团队无法对每个模型都进行精心的手工构建。因此，在生产中最好直接部署数据挖掘技术。如此一来，其关键就是构建预警系统，以将异常情况及时告知数据科学团队并提供失效保护操作（Raeder 等 , 2012）。



我们也可以选择不那么“技术范”的部署方式。在一个著名的案例中，数据挖掘发现了一套能帮助人们快速诊断工业印刷中的常见错误并对其进行修复的规则。而部署这套规则只需要把写有规则的清单贴在打印机旁（Evans & Fisher, 2002）。部署环节也可以更加巧妙，比如改变获取数据的过程，或根据数据挖掘所获得的信息对战略、营销或运营环节做些调整。

将模型部署于生产系统时，往往需要根据生产环境对模型进行重新编码。这通常是为了提高速度或提高该模型与现存系统的兼容性。但这可能会造成高额费用或投入。在许多情况下，数据科学团队不仅需要开发出一个可运行的原型，还需要对其进行评估，然后再将其转交给开发团队进行编码实现。



实际操作中，由数据科学团队建模到转交开发团队实现的过程是有风险的。请谨记这句话：“你的模型不是数据科学家设计的那个，而是数据工程师搭建的那个。”从管理层角度看，开发团队最好尽早派成员参与到数据科学项目中，以顾问的身份向数据科学团队提供意见和建议。在实践中，这类特殊的开发人员实质上逐渐变成了“数据科学工程师”，即在生产系统和数据科学两方面都拥有专业知识的软件工程师。随着项目的推进，他们的责任也愈发重大。有时他们需要取得主导地位，行使对产品的主导权。一般而言，数据科学家们需要自始至终参与项目，直至其最终部署。依据技能不同，他们的身份既可以是顾问，也可以是开发人员。

不管部署环节是否成功，整个流程往往都会再回到商业理解环节。数据挖掘流程能够暴露出商业问题和其解决方案的难点，而通过第二次迭代，就能改进解决方案。单是思考业务、数据和绩效目标的过程，往往就有助于想出提升业绩的新思路，有时甚至还能开辟新的业务线或创造新的投资机会。

值得注意的是，不一定非要等到部署环节失败才能重启数据挖掘的大循环。在评估环节就可能发现评估结果并未达到部署标准，而此时就需要调整问题定义或获取其他数据。这个过程即图 2-2 中由评估环节指向商业理解环节的“捷径”。在实际中，每个环节都应有回到其之前环节的“捷径”，这是因为数据挖掘流程的每个环节都有一定的探索性，而当有新发现需要纳入考量时，我们就需要有足够的灵活性来退回到之前的各环节。[5]

2.5 管理数据科学团队的含义

我们很容易把数据挖掘流程视为软件开发过程，但这是错误的。诚然，数据挖掘项目往往被当成工程项目并受到和工程项目相同的管理。当数据挖掘项目是由软件部门发起时，这还情有可原。毕竟数据是由大型软件系统生成的，而分析结果最终也要反馈进该系统。管理人员通常更熟悉软件技术，且更擅长管理软件项目，因为软件项目的里程碑很容易商定，而项目成功与否通常也很明确。当软件项目的管理人员看到 CRISP 数据挖掘循环（见图 2-2）时，可能觉得它与软件开发循环非常相似，因此他们认为自己如果用管理软件开发项目的方法来管理分析项目，很快就会得心应手。

然而这种想法并不正确，因为数据挖掘是一项探索工作，它更接近于研究和开发，而不是工程。CRISP 循环就基于探索，其迭代的是**方法**和**策略**，而非软件设计。其产出的结果可能非常不确定，而且任何一步的结果都有可能改变对问题的基本理解。直接开发用于部署的数据挖掘解决方案是一个昂贵且不成熟的想法。与此相反，各种数据分析项目往往需要通过在信息上投资来从各方面降低不确定性。我们可以先小规模地投资于试点研究和一次性原型。数据科学家也应通过文献研究寻找其他方案及其具体运作方法。如果团队考虑进行大规模的投资，则可以搭建可供敏捷试验使用的测试平台。如果你是一名软件工程管理人员，那么上述这些内容看起来可能更像研究与探索，而不是你习以为常的工作。而这甚至会让你不太适应。



软件技能与分析技能

虽然数据挖掘与软件相关，但其所需的不仅仅是程序员常用的编程技能。软件工程崇尚按需求编写高效、高质量的代码。在评估其团队成员时，他们也使用软件指标，如该成员所编写的代码量或所修复的故障数。然而对分析师而言，更重要的是能够明确表达问题、迅速构建解决方案、对结构拙劣的问题提出合理假设、设计能够代表大量投资的实验和对结果进行分析。因此，在建立数据科学团队时，以上这些技能（而非传统的软件工程的专业能力）才是需要考虑的。

注 5：对软件方面的专业人员而言，这种情况很像一句哲言：“失败越快，成功越早。”（Muoio, 1997）

2.6 其他分析技巧与技术

商业分析涉及许多技术在数据分析上的应用，其中大部分并非本书的重点内容，本书重点关注的是数据分析式思维以及从数据中获取有用模式的原理。然而，我们仍需要熟悉这些相关技术，了解它们的目的和作用，并且清楚何时应该向相关专家寻求帮助。

为此，本章将展示六组相关的分析技术，并在合适的时机将其与数据挖掘进行比较。两者最主要的区别是，数据挖掘致力于从数据中**自动**寻找**知识**、**模式**和**规律**。[6] 商业分析师的一项重要技能就是识别出适合解决特定问题的分析技术。

2.6.1 统计

“统计”一词在商业分析中有两种用法。第一种是作为从数据中计算特定数值时的万能名词（比如：“我们得收集一些关于顾客使用量的统计数据，看看出了什么问题”）。这些数值通常包括总和、平均值和比率等，此处我们称其为“汇总统计量”。我们往往想挖掘更深处的信息，并按某些特定条件来计算总体中一个或多个子集的汇总统计量（比如：“男女用户的流失比率是否不同？”和“美国东北地区高收入顾客的流失比率如何？”）。汇总统计量是许多数据科学理论和实践中的基本元素。

汇总统计量应根据要解决的商业问题来仔细选择（这是一条基本原则，接下来会讲到），选择时也需要注意进行汇总统计的数据的**分布情况**。比如，根据美国《2004 年人口普查局经济调查报告》，美国人均收入（平均值）超过了 6 万美元，但用这个数据来衡量平均收入、辅助政治决策会造成误导，因为美国人口的收入分布是非常不平衡的：许多人收入极低，某些人则收入极高。在这种情况下，算术平均值所能传达的人口收入信息相对是很少的。因此，应该用另一种指标来表示“平均”收入，如中位数。人口收入的中位数表示在所有人中，有一半挣得比这个数多，而另一半挣得比这个数少。2004 年美国人口普查研究显示，美国人口收入的中位数仅有 44 389 美元，比平均值小得多。这个例子似乎十分浅显，因为我们已经很熟悉“收入中位数”这一概念了，但同样的道理也适用于任何汇总统计量的计算。在开始统计前，不妨问问自己：是否周全地考虑了所要解答或回答的整个问题？是否考虑了数据的分布情况？所选的统计量是否合适？

“统计”一词的另一种用法则是指学科，即常说的“统计学”。统计学中的很大一部分知识构成了分析学的理论基础，而统计学也可以被视为数据科学这个大领域的一部分。比如，统计学能让我们了解不同的数据分布，以及它们各自适用的汇总统计量；统计学还能让我们知道如何使用数据来检验假设和估计结论的不确定性。关于数据挖掘，假设检验可以用于判定数据挖掘所发现的模式是有效而普适的规律，还是在特定数据集中出现的偶然现象。与本书相关的是，许多从数据中获取模型或模式的技术都能在统计学中找到其理论根源。

比如，经过初步研究，可能会发现美国东北部地区的用户流失率是 22.5%，而全美国的平均用户流失率仅为 15%。这种情况可能仅仅是一种偶然的波动，毕竟用户流失率是一个会随地区和时间改变的变量。可美国东北部的流失率是全美平均水平的 1.5 倍，这好

注 6：值得注意的是，完全自动地从数据中获得发现非常罕见。这其中的关键是：数据挖掘至少会将部分模式寻找和知识发现的过程自动化，而非只是为人工的查找过程和发现过程提供技术支持。

像有些过高了。这种情况仅由随机变化导致的可能性是多少呢？统计假设检验就能够回答这类问题。

与此相关的一个概念就是使用置信区间将不确定性量化。尽管全美用户流失率为 15%，但是这个值不是固定的。通过传统数据分析可以得出，在 95% 的情况下，用户流失率在 13% 到 17% 之间波动。

数据挖掘中被称作**假设提出**的过程与上述过程正好互补。我们能否一开始就发现数据中的模式？假设在被提出之后应该经过谨慎的检验（通常基于不同的数据，参见第 5 章）。另外，在数据挖掘中会出现数值估计，而这些数值估计也通常需要给出置信区间。这个话题在随后探讨数据挖掘结果评估时会再行讨论。

本书不会用过多的篇幅来讨论这些基础的统计学概念。因为关于统计学和商业统计的入门书已经非常多，所以如果本书非要讨论某个主题的话，其观点会非常狭隘或非常浅薄。

即便如此，在商业分析背景下，我们还是会经常听到“相关性”这个统计术语，比如：“有什么指标与未来的用户流失有相关性？”就像术语“统计（学）”一样，“相关性”也有一个通用含义（一个数量的改变预示着另一个数量的改变）和一个特定的技术性含义（例如，由特定数学公式定义的线性相关）。相关性的概念将是后文中（从下一章开始）关于商业数据科学余下部分讨论的出发点。

2.6.2 数据库查询

查询是一种由专门语言编写，从数据库系统中请求数据子集或数据统计的操作。许多工具都可以用来执行分析人员发出的一次性或重复性的数据请求。这类工具通常是数据库系统的前端，我们可以基于结构化查询语言（SQL）或图形用户界面（GUI）建立查询（如实例查询，也称 QBE）。比如，如果分析师能定义一个可基于数据库内数据计算的操作术语“盈利性”，那么查询工具就可以回答“谁是美国东北部地区带来最多利润的用户”这个问题。运行查询之后，分析师会得到一个按带来利润的多少排序的用户名单。由于查询本身不会发现模式或者模型，所以它与数据挖掘在本质上是不同的。

数据库查询适用于分析师清楚数据中的哪个子集值得分析，并打算研究这个子集或验证某个关于它的假设的情况。比如，如果分析师怀疑美国东北部地区的中年男性存在一些特别值得关注的用户流失行为，他就可以编写如下 SQL 查询语句：

```
SELECT * FROM CUSTOMERS WHERE AGE > 45 and SEX='M' and DOMICILE = 'NE'
```

如果我们希望针对这些用户投放优惠活动，那么查询工具就能从数据库的 `CUSTOMERS` 表中找到他们的所有信息（用“*”来选择）。

与此相反，我们可以首先用数据挖掘（以数据中的模式或规律的形式）编写这条查询。数据挖掘过程可以先检查之前的用户流失状况，然后判定可以对该部分（年龄大于 45 岁，性别为男性，居住地为美国东北部）用户的流失率做出相应预测。这个标准被转化成 SQL 语句后，查询工具就能够在数据库中找到符合要求的记录。

查询工具通常能够执行复杂的逻辑运算，包括计算数据子集的汇总统计、排序、用相关数据关联多个数据表等。数据科学家往往非常擅长通过编写查询语句来获取所需数据。

为了辅助数据探索，联机分析处理（OLAP）提供了一个易于使用的图形用户界面来查询大型数据集。“联机”是指该处理过程是实时的，由此分析师和决策者可以快速高效地得到查询结果。与 SQL 之类查询工具的“临时”查询不同，用于 OLAP 的分析维度都需要预先编码写入系统。如果我们预料到要探索销量与地区和时间的关系，那么就需要提前把这三个维度编入系统，随后简单地通过点击、拖动和操作动态表格的方式下钻到总体中。

OLAP 系统的设计目的是帮助分析师实现对数据的人工和可视化探索，它并不能进行建模或自动发现模式。而数据挖掘能够在数据探索过程中轻松地将新的维度纳入分析中。不过，OLAP 工具可以作为数据挖掘工具的有力补充，帮助探索商业数据。

2.6.3 数据仓库

数据仓库可以从整个企业（通常是从多个拥有独立数据库的交易处理系统）中收集数据并进行合并，以供分析系统访问。数据仓库可以被视作数据挖掘工作的辅助工具，但由于大部分数据挖掘工作并不使用数据仓库，所以它并非数据挖掘工作的必备项。然而，使用数据仓库的公司往往能够将数据挖掘进行得更广泛、更深入。比如，如果数据仓库不仅包含人力资源数据，还包含销售数据和收银数据，就可以用来探索优秀销售人员的特征模式。

2.6.4 回归分析

本书中讨论的一些方法是另外一套分析方法的核心，后者通常被归为**回归分析**，并且被广泛地应用于统计学及其他基于计量经济分析的领域。相比一般的介绍回归分析的教材或课程，本书的侧重点有所不同。本书不会解释特定数据集，而更关心如何从中获取适合推广的模式，以便改进相关的商业流程。通常，这会涉及估计和预测未在已分析的数据集内的目标变量的值。举个例子，在本书中，比起根据某组特定的历史数据深入挖掘用户流失的原因（尽管它很重要），我们更想预测现存用户中哪些是预防用户流失的最佳目标。因此，本书将花些篇幅来讨论如何通过使用新数据来检验某个模式是否具有普遍意义，以及如何减少某一模式仅适用于某组数据，但不能推广到数据总体的情况。[7]

虽然有关解释性建模和预测性建模的话题会引发深刻的探讨，[8] 但这远超出了本书范围。必须要了解的是，尽管两种建模方法所用的技术有很多重叠，然而解释性建模所得出的内容不全适用于预测性建模。因此学习过回归分析的读者可能会遇到新知识，甚至与已有知识似乎相矛盾的知识。[9]

2.6.5 机器学习与数据挖掘

机器学习方法是从数据中提取（预测性）模型的一系列方法。它在多个领域同时得到发展，而这些领域中最广为人知的是机器学习、应用统计和模式识别三个领域。其中，机器

注 7：即模型的泛化能力差。——译者注

注 8：感兴趣的读者不妨读一读 Shmueli（2010）的文章。

注 9：读者可以通过深入学习解决这些表面上的矛盾，不过这样的深入学习对于理解数据科学的基础原理而言并非必需。

学习这一研究领域最初是作为人工智能的子学科出现的。而人工智能则致力于依据智能代理[10]的经验，逐步提高其知识水平或性能。这个提高过程通常涉及对环境中的数据进行分析和对未知量进行预测，而这些年机器学习中的数据分析部分在该领域发挥了重要作用。随着机器学习方法被广泛应用，机器学习、应用统计和模式识别这些学科变得关系密切，而各学科之间的界限也变得不那么明显了。

数据挖掘（或“知识发现和数据挖掘”，KDD）起初是机器学习的一个分支，它在后来的发展中仍与机器学习保持着密切的关系。这两个学科不但都涉及通过分析数据来找到有用的或富含信息的模式，而且它们也共享很多技术和算法。两者的关系如此密切，以至于研究者能够同时进行两个领域的研究，并在两者之间自如转换。尽管如此，然而我们仍需指出两者的一些区别。

一般而言，由于机器学习与许多提高性能的方法有关，所以它涵盖了一些不属于 KDD 的子领域，如机器人学和计算机视觉。机器学习还涉及**代理**和**认知**，即智能代理如何运用所学到的知识在其所处环境中进行推断和行动。然而，这些并不是数据挖掘所关注的。

历史上，KDD 作为机器学习的一个分支领域，主要研究现实世界的应用场景中所产生的问题；而十五年后的现在，KDD 与现实应用的联系反而比与机器学习的联系更加密切。在此情况下，对商业应用和商业数据分析问题的研究也更多地被归为 KDD 的课题而非机器学习的课题。KDD 还越来越倾向于关注数据分析的整个流程，如数据准备、模型学习和模型评估等。

2.6.6　运用以上技术解决商业问题

为了演示如何将本章讲述的技术应用在商业分析中，请思考以下可能遇到的问题，并想一想应该运用哪些技术回答这些问题。这些问题虽然联系紧密，但彼此之间仍有细微区别。只有理解了这些区别，才能知道针对这些问题应该使用什么技术，以及在必要时应该向哪些人咨询。

(1) **谁是盈利性最高的用户？**

如果能根据现有数据对“盈利性”进行明确定义，那么这就是一个简单的数据库查询问题。我们可以使用一个标准的查询工具从数据库中提取一组用户记录。其结果可以根据累计交易额或其他盈利性业务指标进行排序。

(2) **盈利性用户和普通用户之间是否真的存在区别？**

这是一个推断问题或假设问题（即假设“用给公司带来的价值来衡量，盈利性用户和普通用户之间确实存在区别”）。我们既可以用统计假设检验来证实或推翻这个假设，也可以用统计分析来推导差异的置信区间或其真实存在的概率。其结果通常为以下形式：“盈利性用户的价值与普通用户的价值存在显著区别。这个区别由偶然因素导致的概率小于 5%。”

注 10：机器人、智能软件和机器设备等。——译者注

(3) **这些用户是谁？他们的特征是什么？**

通常我们不仅想列出这些盈利性用户的名单，还想找出这些用户的常见特征。个体用户的特征或汇总统计量可以通过数据库查询技术从数据库中提取。但更深度的分析应该能够判断出哪些特征可以用来区分盈利性用户和非盈利性用户，这就进入了数据科学的范畴——使用数据挖掘技术实现模式的自动发现。接下来的章节将对此进行深入探讨。

(4) **某位特定的新用户是否能带来利润？根据预期，该用户能带来多少收益？**

这些问题可以利用数据挖掘技术，通过调查历史用户记录并建立盈利预测模型来解决。这样的技术能够通过历史数据产生模型，而模型又能应用于新用户来进行预测。这也是接下来几章的主题。

注意最后两个数据挖掘问题有细微的区别。第一个是分类问题，可以表述为“某个特定新用户是否能带来盈利”（是 / 否或者其概率的问题）。第二个则可以表述成“预测用户能带给公司的价值（数值）”。本书后面会继续深入探讨这两个问题。

2.7　小结

数据挖掘是一门手艺。像其他手艺一样，它有一个定义明确的流程，有助于更容易地取得成功。该流程是面对数据科学项目时关键的概念性思维工具，后文会反复提及这个流程，并展示每个数据科学基本概念如何与它融为一体。反过来，对数据科学基本概念的理解也能大大地提高那些借助了数据挖掘的企业的成功率。

与数据科学相关的各种研究领域发展出了一系列典型的数据科学任务，如分类、回归和聚类。每种任务用途不同，也各有一套与之关联的解决方案。在着手解决一个新项目时，数据科学家通常先将其分解成一或多个基本任务，随后逐一选择这些任务的解决方案，最后再将所有解决方案进行组合。把这个过程做好需要大量的经验和技巧。要想成功开展数据挖掘项目，就要明智地在数据的能力（如数据能预测什么，预测精度如何）和项目的目标之间保持平衡。为此，我们需要牢记数据挖掘结果的使用方法，并使用它为数据挖掘流程本身提供指导。

数据挖掘与统计假设检验和数据库查询（另有专门教材及课程）等重要的支持技术不同，但又与这些技术互补。尽管数据挖掘与相关技术的界限有时并不明显，但仍需了解其他技术的用途和优势，以确定何时需要使用它们。

对业务管理者而言，数据挖掘流程是一个用于分析数据挖掘项目或提案的有效框架。该流程可以将分析系统地组织起来，其中所包含的一系列问题，则可以用来帮助检验项目或者提案是基于良好的构思，还是有根本缺陷。本书会在详细讨论更多的基本原理后，再回顾这一部分。

第3章

预测建模导论：从相关性到有监督的划分

基本概念：富信息属性识别；通过逐步属性选择划分数据

示例方法：相关性度量；属性 / 变量选择；树型归纳

前两章概述了模型和建模的概念，本章将深入研究数据科学中的一个重要课题：预测建模。本章将接着使用 1.3 节的数据挖掘示例。首先本章会把预测建模视为**有监督**的数据划分，也就是根据某个值得关注的量，将整个总体划分为不同的群组。具体来讲，就是根据某个希望预测或估计的值对总体进行分组。预测的目标可以是某个想避免的事件，比如哪些用户合约期满时会流失、哪些账户遭受了诈骗、哪些潜在用户会无法结清账户（即不良贷款，如电话账单或信用卡账单的违约）或哪些网页的内容会令人不适等；预测目标也可以是希望发生的事件，比如哪些用户最可能响应某个广告或优惠活动，以及哪些网页最符合某个搜索请求。

在探讨有监督的数据划分的过程中，本章将引入数据挖掘的一条基础理念：寻找或选择数据所描述的实体的重要且富含信息（“富信息”）的变量或“属性”。虽然“富信息”的含义要视应用场景而定，但一般而言，**信息是能够降低某事件不确定性的量**。比如，假设有个老海盗把关于他藏宝地点的信息告诉了我，这并不意味着我确切地知道宝藏的所在位置，而仅仅意味着对我而言，藏宝地点的不确定性降低了。而告诉我的信息的质量越高，这种不确定性就越小。

现在请读者回想一下前一章中谈到的数据挖掘中“有监督”的概念。进行有监督的数据挖掘的关键，是要有一个想要预测的或希望更深入地理解的目标变量。而该变量在真正需要决策时往往是未知或不可知的，比如某个用户是否会在合约到期后很快离开，或哪个账户遭受了欺诈。目标变量能够让我们更清晰地了解什么是“寻找富信息属性”，即是否存在一个或多个能够减小目标变量的不确定性的变量。同时，关于上文所讨论的相关性的一般性概念，这里给出了一个常见的分析型应用：我们希望找到与目标变量相关的可知属性，以减小该目标变量的不确定性。而仅是寻找这些相关变量的过程本身，就有助于更加深入地理解这个商业问题。

寻找富信息属性有助于处理体量日益庞大的数据库和数据流。当需要对过于庞大的数据集进行分析时，计算将成为一个巨大的挑战。在分析师缺少高性能计算机的情况下，这个问题尤为突出。针对这个问题，一个经过实践检验的解决方法就是先从数据集中选出一个子集来分析。而选择富信息属性则为选择富含信息的数据子集提供了一种“聪明”的办法。另外，如果在数据驱动建模前先选择变量，也有助于提升建模的精度，本书会在第 5 章探讨其原因。

寻找富信息属性也是一种被称作**树型归纳**的预测模型的基础。该模型应用广泛，3.6 节将把它作为预测模型这一基本概念的一项应用加以介绍。树型归纳能通过一种巧妙的方式，即不断重复选择富信息属性，对数据进行有监督的划分。学完本章，你将能够理解：预测建模的基本概念、寻找富信息属性的基本概念和一项具体的演示性实践技术、树形结构模型的基本概念，以及从数据集中获取树形结构模型的流程（即实施有监督的数据划分的过程）。

3.1 建模、归纳与预测

一般而言，模型就是一种为特定目的服务的、简化了的对现实世界的表现。这种简化往往基于某些假设（也就是对上述特定目的而言，哪些问题重要，哪些问题不重要），但有时也基于信息或处理方面的限制。例如，地图就是真实世界的一个模型。制图师舍去了大量与地图目的无关的信息，仅仅保留与其目的相关的信息，有时甚至还会进一步简化它们。比如，公路图仅会保留和突出每条公路、公路的基本拓扑结构、公路与旅行目的地的关系，以及其他相关信息。各行业中都有不同种类的著名的模型，如建筑蓝图、工程原型和 Black-Scholes 期权定价模型等。它们都舍弃了与主要目的无关的细节而仅保留了相关的信息。

在数据科学中，预测模型是一种用来预测我们感兴趣的未知值（即目标变量）的公式。这个公式既可以是数学表达式，也可以是逻辑表达式（如规则），但通常表现为两者的混合体。由于我们把有监督型数据挖掘分为分类和回归两大类，故而接下来也将分别考虑分类模型（以及类概率估计模型）和回归模型。



术语：预测

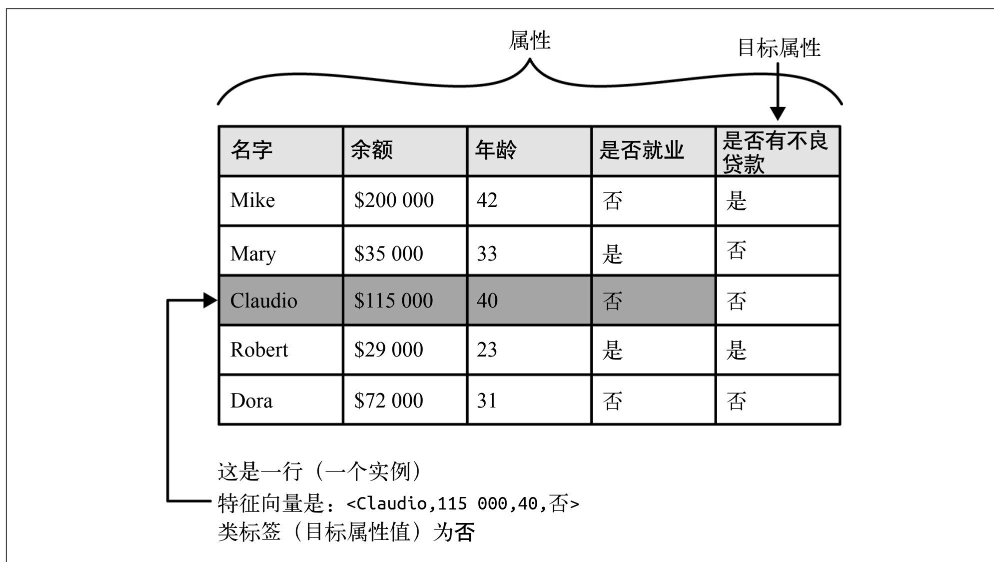
通常，预测是指预报一个未来要发生的事件，而在数据科学中，其更常见的含义是**估计一个未知量**。该未知量既可以是未来发生的事件（即通常含义的“预测”），也可以是当前或过去发生过的事件。实际上，由于数据挖掘所处理的通常是历史数据，所以模型的建立和验证往往也是基于历史事件的。例如，信用评分的预测模型估计的是潜在的用户违约（即产生不良贷款）风险；垃圾邮件过滤预测模型估计的是某封邮件是否为垃圾邮件；欺诈检测的预测模型判断的则是某个账户是否遭受了欺诈。关键在于，预测模型所估计的是某个未知量。

这样看来，预测建模与**描述**建模截然不同。后者的主要目的不是估计某个值，而是试图了解某个现象或过程背后的本质。比如，用户流失行为的描述模型可以告诉我们，流失的用户具有哪些典型特征。[1] 在某种程度上，描述模型的评估标准是其可理解性，我们可能倾向于选择一个精度不够高，但比较好理解的模型。而对于预测模型而言，可理解性固然很重要，但其评估标准却是预测能力。这两种模型的区别并没有以上所说的那么严格：它们会共用某些技术，而且一个模型通常可以兼顾预测和描述两个目的（尽管有时效果欠佳）。有时候，预测模型的价值不在于预测结果本身，而主要在于观察预测模型时所获得的对问题的理解。

在深入讨论预测模型之前，有必要先引入一些术语。有监督学习是一个建立模型的过程，该模型描述了一系列所选变量（**属性**或**特征**）和一个预先确定的变量（**目标**变量）之间的关系。预测模型就像是特征变量的函数（一般是概率函数），被用来估计目标变量的值。因此，在用户流失预测问题中，可以建立一个用户流失倾向模型，即一个函数。其自变量可以是用户账户的属性，如年龄、收入、就业时间、呼叫客服次数、超额费用、用户地理分布、数据使用量，等等。

图 3-1 通过展示一个极简的信贷不良贷款预测示例，阐释了刚刚介绍的一些术语。一个**实例**或**示例**表示一个事件或一个数据点，在此例中即为一个曾被发放信贷的历史用户；在数据库或电子表格中这也被称为**一行**。一个实例由一系列**属性**（又称域、列、变量或特征）所描述。因为它可以表示为一组长度固定且有序的特征值（向量），所以有时候实例也被称为**特征向量**。除非特别声明，否则本书将默认数据中所有属性都有相应的值（目标变量除外）。

注 1：描述建模通常用来帮助人们理解数据产生过程中的因果关系（如：**用户为什么流失**）。



名字	余额	年龄	是否就业	是否有不良贷款
Mike	$200 000	42	否	是
Mary	$35 000	33	是	否
Claudio	$115 000	40	否	否
Robert	$29 000	23	是	是
Dora	$72 000	31	否	否

图 3-1：有监督分类问题的数据挖掘术语。一个问题之所以是“有监督”的，是因为该问题中含有一个目标变量，以及一些目标变量值已知的“训练”数据。它之所以是一个分类问题（而非回归问题），是因为其目标变量是类别型的（是或否）而非数值型的

一物多名

历史上，由于许多不同领域（包括机器学习、模式识别、统计学和数据库等）都对数据科学的原理和技术进行了研究，所以数据科学中的同一个概念往往具有多个名称。人们常说的**数据集**，其形式与数据库中的**表**和电子制表软件中的**工作表**是一致的。数据集包含一系列**示例**或**实例**，而实例既是数据库中的表里的**行**，也是统计学中的**案例**。

特征（即表中的列）也有许多不同的名称；在统计学中，作为输入，属性被称为**独立变量**或**预测变量**，在运筹学研究中则叫作**解释变量**；目标变量，因为其值需要被预测，所以在统计学中常被称为**依赖变量**。这种命名方式很容易造成混淆，因为独立变量不一定相互独立（或与其他元素独立），而依赖变量也不一定总是依赖于所有的独立变量。因此，本书回避了这种命名方式。一些专家认为目标变量也属于特征，另一些人则不这么认为。但有一点显然非常重要：目标变量不能用来预测它自己。不过预先给定的目标变量的值，会对预测未来的目标变量有巨大的帮助，因此这些预先值也可以被视作特征。

根据数据建立模型的过程也叫模型归纳。归纳是一个哲学术语，表示将具体案例推广为一般性规则（或规律、真理）。既然模型就是统计意义上的一般性规则（一般来说，它们并非 100% 正确，有时其正确率很低），那么根据数据进行建模的程序就叫作归纳算法或学习器。大多数归纳过程被转化为分类模型或回归模型。不过由于分类在统计学的其他领域中被讨论得较少，而它又与许多商业问题密切相关（因此数据科学中的许多工作都聚焦于分

类），所以后文将主要讨论分类问题。



术语：归纳和演绎

与归纳相对的概念叫作**演绎**。演绎可以从一般性规律和具体事实出发，推演出其他具体事实。**使用**模型的过程就是一个（概率）演绎过程。本书将很快讲到这一点。

归纳算法所输入的数据被称作**训练**数据，可以用来归纳出模型。第 2 章中提到过，由于训练数据中的目标变量（即标签）的值已知，所以训练数据也叫**标注**数据。

现在回到用户流失示例中。根据在第 1 章和第 2 章学到的知识，在建模环节我们可能会想建立一个“有监督的划分”的模型，依据合约期满后流失（平均）概率高低，把样本数据划分为两部分。至于如何做到这一点，请读者思考本书的基本概念之一：如何选择出一个或者多个属性 / 特征 / 变量作为依据，尽可能地把样本数据**按照我们感兴趣的目标变量**进行划分？

3.2　有监督的划分

预测模型主要用来估计我们关心的目标变量的值。用有监督方法获取数据中所含模式最直观的方法，就是尝试把总体划分成目标变量值不同的子群（同时让子群内的目标变量值相近）。如果在目标变量值未知时，能知道用哪些变量值来做上述数据集划分，那么这样的划分就可以用来预测目标变量值。而且，这样的划分还能提供一系列很好理解的划分模式。举例说明：“居住在纽约市的中年专业人士的平均用户流失率为 5%”，其中“居住在纽约市的中年专业人士”是划分的标准（表示某些特定属性），“流失率为 5%”则是该划分中目标变量的预测值。[2]

当问题中有很多属性，却不确定如何划分数据时，我们往往会倾向于应用数据挖掘。在用户流失的预测问题中，用来预测流失倾向的最佳划分是未知的。假如数据中真的有某种划分方法，能将目标变量划分为（平均值）明显不同的几类，我们就需要找到自动获取这种划分方法的办法。

这就引出了本书的基本概念：如何判断某变量是否包含关于目标变量的重要信息？如果包含，那么信息量有多大？我们希望能自动选择和手头任务有关的、信息量更大的变量（换言之，即预测目标变量值）。更进一步，我们甚至希望可以按照预测目标变量的准确程度对这些变量进行排序。

现在，本书仅考虑如何选择信息量最大的那个富信息属性。本书将通过解决这个问题来介绍第一项具体的数据挖掘技术。该技术虽然简单，却易于扩展且非常有用。在用户流失示例中，关于未来总体中的用户流失率，哪个变量提供的信息最多？专业人士的身份？年龄？住所？收入？向客服投诉的次数？还是超额费用数额？

注 2：接下来本书将讲到，有多种基于数据的方式可用来预测目标变量值，而现在读者可以先大致把它视为训练数据集中划分到每个分组的某种平均值。

接下来，本书将仔细研究一种选择富信息属性的有效方法。在此之后本书会展示如何通过重复地使用该技术来构建有监督的数据划分。尽管直接使用多个变量进行有监督划分的方法非常有用且易于理解，但请记住，它仅是选择富信息变量这个基本观点的一种应用而已。而这个概念应该成为一项理论工具，以便于在更加广泛的层面上思考数据科学。比如，之后本书会逐步研究其他建模方法，而这些方法中并不直接包含变量选择。当你面对的数据集包含极多的属性时，不妨回顾并应用这条极其有用的概念来选出一个富信息属性子集。这样做不仅可以将庞大笨重的数据集大幅度缩小，而且我们往往会发现，从中生成的模型的精度会随之显著提高。

3.2.1 选取富信息属性

如果给定一个大的示例集，那么应该如何选择一个属性，使得依据它进行划分之后，数据集所含的信息量最大呢？思考一个具体的二元（两类）分类问题，想一想从中能够得出什么。在如图 3-2 所示的简单的数据划分问题中，图中的 12 个火柴人脑袋有两种形状（方形和圆形），而他们的身体不仅有两种形状（长方形和椭圆形），还有两种颜色（灰色和白色）。

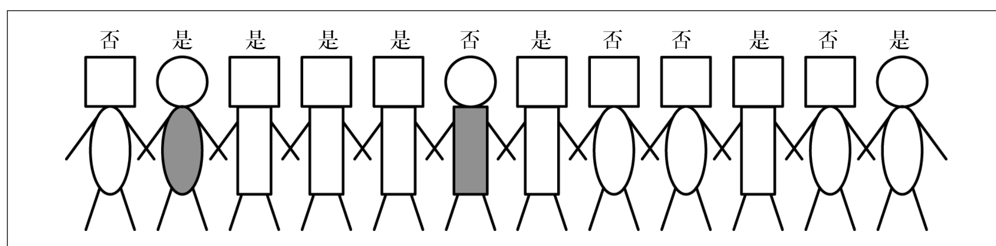


图 3-2：图中所展示的是一群需要分类的人。他们头顶的标签为目标变量值（是否有不良贷款），而他们身体或脑袋的颜色和形状则代表不同的预测变量属性

我们将用以上这些属性来描述这些人。他们头顶上的二元（**是**或**否**）标签表示此人是否有不良贷款。我们可以像下面这样描述有关这群人的数据。

- 属性
  - 脑袋形状：方形 / 圆形
  - 身体形状：长方形 / 椭圆形
  - 身体颜色：灰色 / 白色
- 目标变量
  - 是否有不良贷款：是 / 否

试问：哪个属性能最明确地把这些人中有不良贷款的和无不良贷款的划分开？我们想在结果中看到尽可能**纯**的分组。这里的**纯**是指**目标变量值的同质性**。如果一个组内所有成员的目标变量值都相同，那么该分组就是纯的；如果组内至少有一个成员的目标变量值与其他成员不同，那么该分组就是不纯的。

可惜在现实中极少能找到能把数据划分得绝对纯的变量。但是，只要能显著降低分组的不

纯度，就可以对数据（以及与之对应的总体）多一分了解。而对于本章而言，更重要的是我们还能将该属性应用于预测模型，比如在该示例中，我们需要预测其中一组出现不良贷款的可能性与另一组相比孰高孰低。如果可以做到这一点，我们就可以为预测结果中产生不良贷款可能性低的用户提供信用贷款，或根据用户产生不良贷款的可能性大小来给他们提供不同额度的信用贷款。

从技术角度来看，分类问题有几个复杂之处。

(1) 我们很难根据各种属性将数据集完美地划分开来。即便有一个子组碰巧是纯的，其他的也未必纯。比如，假设图 3-2 中的第二个人不存在，那么**身体颜色 = 灰色**这个部分就是纯的（**是否有不良贷款 = 否**），但随之产生的另一个部分**身体颜色 = 白色**，却仍不纯。

(2) 在上一个例子中，条件**身体颜色 = 灰色**通过去掉一个数据点而分出了纯子集。倘若存在另一种划分方式，不产生任何纯子集，却能更广泛地降低各个子集的不纯度，那么这两种划分方式孰优孰劣？

(3) 不是所有属性都是二元的；有的属性存在三个或更多不同的值。必须考虑到：某个属性可以将数据集分为两个子集，而另一个属性则可能将数据集分为三个甚至七个子集。这样一来，我们如何比较这些子集呢？

(4) 某些属性是数值型的（连续的或整数的）。那么对每个数值都进行划分是否有意义？（答案是否定的。）这种情况下，要如何对数值型属性进行有监督的划分？

幸运的是，对于分类问题而言，我们可以通过一个公式来解决上述所有问题。该公式可以用于测量基于每个属性所进行的划分的好坏程度（对特定目标变量而言）。这个公式的功能称作**纯度测量**。

最常见的划分数据的指标被称为**信息增益**，它基于一个被称作**熵**的纯度测量指标。以上两个概念皆由信息论先驱 Claude Shannon 首创，其作品在该领域具有开创性地位（Shannon, 1948）。

熵可以用于测量集合中的无序程度，见上文例子中的个体的划分问题。试想，一个集合中的各个成员具有一组**性质**，每个成员有且只有这组性质中的一种。在有监督的划分中，成员的性质就相当于目标变量的值。混乱则指的是某个分组中这些性质的混合（或不纯）程度。所以，打个比方，一个混合了许多有不良贷款者和无不良贷款者的分组就具有较高的熵值。

严格意义上，熵定义如下。

公式 3-1：熵

$$\text{熵} = -p_1 \log(p_1) - p_2 \log(p_2) - \cdots$$

其中，p_i 是集合中性质 i 的概率（相对百分比），其取值范围是 0 到 1（1 表示集合中所有成员都有性质 i，0 则表示所有成员都没有性质 i）。省略号仅表示性质可能多于两个（在技术领域中，通常取 2 作为对数的底）。

熵的公式本身可能不太直观，因此难以理解。图 3-3 展示了一个测量熵的例子，其中每个集合包含 10 个二元分类（“+”和“–”）实例。从中可以看出，从 0 到 1，熵的值测量的是集合的整体混乱程度。其中 0 代表最小的混乱程度（集合中所有个体的性质相同），而

1 代表最大的混乱程度（即不同的性质均匀地混合）。由于这是二元分类，因此 $p_+ = 1 - p_-$。从左下角全为“–”的集合看起，该集合满足 $p_+ = 0$，混乱程度达到最小（极纯），熵值为 0。当开始把集合中个体的类标签从“–”转换为“+”时，熵值也随之增加了。当集合中的个体的类别分布均衡（即“+”“–”各 5 个）时，熵值达到最大，这时 $p_+ = p_- = 0.5$。随着更多的标签发生转换，“+”开始占多数，此时熵值再度减小。到所有个体都为“+”时，$p_+ = 1$，熵值又达到了最小值 0。

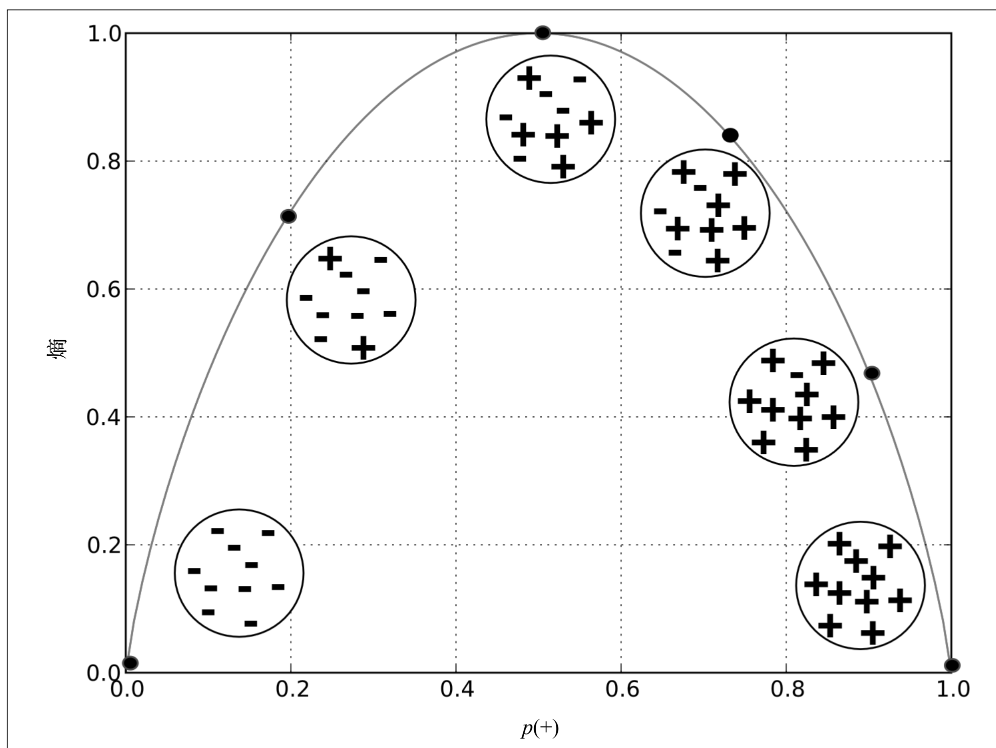


图 3-3：二元集合的熵用 $p(+)$ 的函数表示

举一个具体例子，试想，集合 S 中有 10 个人，其中包含 7 个**无不良贷款者**和 3 个**有不良贷款者**，那么：

$$p\,(\text{无不良贷款}) = 7/10 = 0.7$$

$$p\,(\text{有不良贷款}) = 3/10 = 0.3$$

$$\begin{aligned}\text{熵}(S) &= -[0.7 \times \log_2(0.7) + 0.3 \times \log_2(0.3)] \\ &\approx -[0.7 \times (-0.51) + 0.3 \times (-1.74)] \\ &\approx 0.88\end{aligned}$$

熵的定义和计算只是所要讨论的一部分，我们想要知道的是如何测量属性的（关于目标变量的）信息量，即该属性能给（关于目标变量的）信息量带来多少提升。一个属性可以将一个实例集合划分为几个子集，而熵却只能告诉我们单个子集的不纯度。幸运的是，在用

熵测量任意一个集合混乱程度的基础上，可以定义**信息增益**（IG）这一概念，并用它来测量一个属性（在依据其所做出的划分中）对熵值提高（降低）的影响。严格地讲，信息增益测量的是加入新信息后熵值的**改变**；而在有监督的划分的情形中，我们考虑的却是根据单一属性对数据集进行划分后的信息增益。假设用于划分的属性有 k 个不同的值，记原集合为**父集**，则划分后得到 k 个**子集**。从这个角度看，信息增益就是父集和子集的函数——该属性提供了多少信息量？这取决于子集的纯度相对于父集提高了多少。在预测模型的语境下讲，就是：如果知道了一个属性的值，那么这个信息能在多大程度上提高对目标变量值的认识？

信息增益的明确定义如下。

公式 3-2：信息增益

$$IG(\text{父集，子集}) = \text{熵}(\text{父集}) - [p(c_1) \times \text{熵}(c_1) + p(c_2) \times \text{熵}(c_2) + \cdots]$$

显然，每个子集（c_i）的熵的权重是各子集中所含实例数的比例 $p(c_i)$。这对上文的问题做了回答：与其去除一个实例以制造一个纯子集，不如把父集分为两个比较大的、相对较纯的子集——哪怕这两个子集都不纯。

图 3-4 展示了一个二元分类问题（●和★）。以之为例，图中的子集看起来显然比父集更“纯”。父集中含有 30 个实例，包括 16 个“●”和 14 个“★”，所以：

$$\begin{aligned}\text{熵}(\text{父集}) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\bigstar) \times \log_2 p(\bigstar)] \\ &\approx -[0.53 \times (-0.9) + 0.47 \times (-1.1)] \\ &\approx 0.99(\text{非常不纯})\end{aligned}$$

左侧子集的熵是：

$$\begin{aligned}\text{熵}(\text{账户余额} < 5\text{万}) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\bigstar) \times \log_2 p(\bigstar)] \\ &\approx -[0.92 \times (-0.12) + 0.08 \times (-3.7)] \\ &\approx 0.39\end{aligned}$$

而**右侧**子集的熵是：

$$\begin{aligned}\text{熵}(\text{账户余额} \geqslant 5\text{万}) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\bigstar) \times \log_2 p(\bigstar)] \\ &\approx -[0.24 \times (-2.1) + 0.76 \times (-0.39)] \\ &\approx 0.79\end{aligned}$$

由公式 3-2 可知，该划分的信息增益是：

$$\begin{aligned}IG &= \text{熵}(\text{父集}) - [p(\text{账户余额} < 5\text{万}) \times \text{熵}(\text{账户余额} < 5\text{万}) \\ &\quad + p(\text{账户余额} \geqslant 5\text{万}) \times \text{熵}(\text{账户余额} \geqslant 5\text{万})] \\ &\approx 0.99 - [0.43 \times 0.39 + 0.57 \times 0.79] \\ &\approx 0.37\end{aligned}$$

因此该划分能大幅降低熵。用预测模型的术语说，该属性提供了大量有关目标变量值的信息。

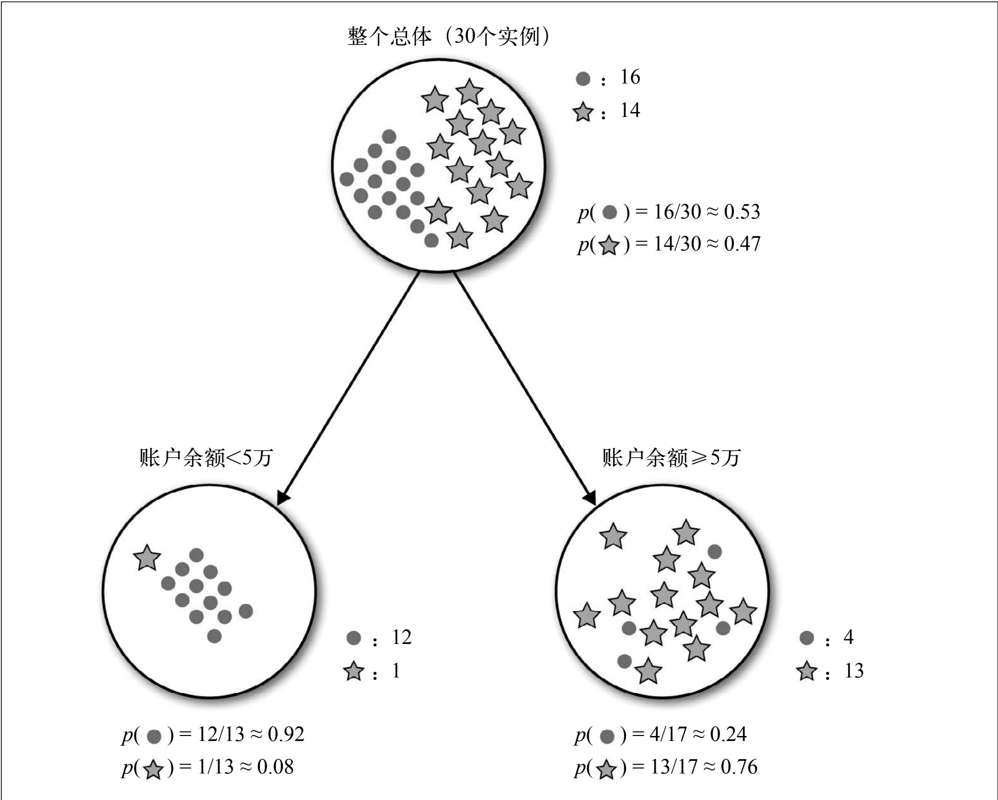


图 3-4：根据（账户余额）账户余额是否少于 5 万，将“有不良贷款”样本的数据分为两组

再举一个例子。图 3-5 候选人分类示例中的父集与图 3-4 中的相同，但现在我们考虑的是按“居住方式”将数据集分为三类：自有、租赁及其他。熵的计算结果如下：

$$\text{熵 (父集)} \approx 0.99$$

$$\text{熵 (居住方式 = 自有)} \approx 0.54$$

$$\text{熵 (居住方式 = 租赁)} \approx 0.97$$

$$\text{熵 (居住方式 = 其他)} \approx 0.98$$

$$IG \approx 0.13$$

变量“居住方式”的信息增益虽然的确为正，但比变量“账户余额”的信息增益要低。直观地看，这是因为，虽然子集“自有”显著降低了熵值，但另外两个子集“租赁”和“其他”的纯度却较父集有所下降。因此基于这些数据，变量“居住方式”比“账户余额”所含的信息量要少。

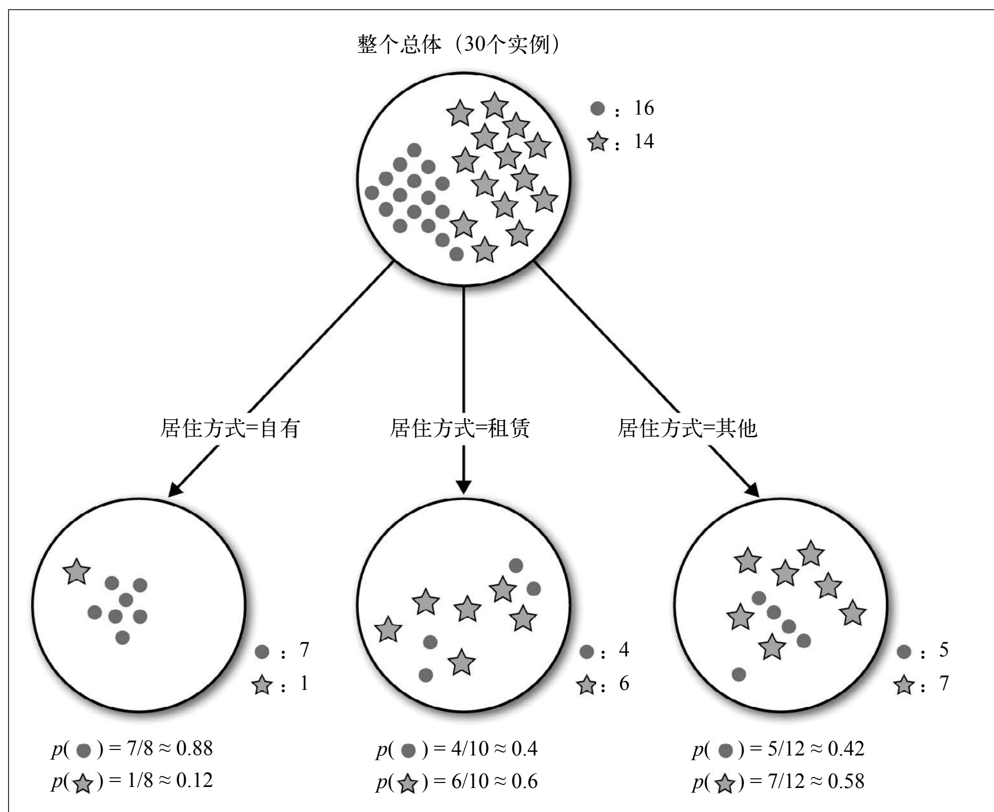


图 3-5：根据三值属性“居住方式”产生的分类树

我们在前文中对分类问题进行有监督的划分的所有担心，都随着信息增益的出现而化解。信息增益不追求绝对的纯度，而且可以应用在子集为任意数量的问题上。它还考虑到了子集的相对规模问题，给予了规模较大的子集相对较大的权重。[3]



数值型变量

我们还未讨论如何处理数值型的属性。我们可以将数值型变量“离散化”，即选择一个（或多个）划分点划分数值，然后将划分后的结果作为一个类别型变量的问题来处理。例如，收入可以被划分成两个或多个区间。我们可以用信息增益来测量数值型变量离散化后的划分结果。不过，如何选择数值型属性离散化的划分点的问题尚未解决。理论上，可以尝试所有合理的划分点，最终选出信息增益最大的一个点。

注 3：严格来讲，这里仍有一个问题：当用于划分的属性含有很多不同值的时候，可能会出现某种划分方法的信息增益很大但预测效果却很差的情况。这个问题（“过拟合”问题）是第 5 章的主题。

最后，如何对回归问题进行有监督的划分呢？这个类型的问题的目标变量可是数值型的！虽然我们仍然需要关注如何降低子集的不纯度，但信息增益这个测量指标已不适用于这种情况。这是因为它是基于熵得出的，即根据不同性质在划分结果中的分布计算出的。因此，需要寻找一个能够度量子集中数值型变量的纯度的方法。

方差就是一个专门用来衡量数值型变量不纯度的指标。本书在此省略对这个指标的算法的推导，读者目前只需记住，这个基本概念非常重要。如果子集内所有个体的目标变量值都相同，那么该子集就是纯的，这时方差为零；如果子集内所有个体的目标变量值差别很大，那么该子集的方差就会非常高。我们可以把父集和子集的方差的减少量当作一个类似于信息增益的指标来使用，这个处理过程完全可以类比上文中信息增益的推导过程。给定一个数值型目标变量，其最佳划分应使得加权平均方差减小的幅度最大。这其实意味着我们仍需要找到与目标变量关系最密切的变量，换句话说，也就是要找到最具预测性的变量。

3.2.2 示例：基于信息增益进行属性选择

现在我们做好了准备，可以应用第一项具体的数据挖掘技术了。一个数据集之中的每个实例都由若干属性和一个目标变量描述，我们可以判断哪个属性对于准确估计目标变量值而言是信息量最大的（将在下文中深入探讨）。我们还可以根据这种信息量，尤其是根据它们的信息增益，对属性进行排序。因此它不仅有助于更好地理解数据，或者预测目标变量，还有助于在不想或无法处理全部数据集的时候，选出一组属性以减小数据集的规模。

为了演示信息增益的用法，本章将使用一个简单但真实的数据集。它来自加州大学欧文分校机器学习数据仓库[4]。该数据集取自“The Audubon Society Field Guide to North American Mushrooms”，其中包含了可食用蘑菇和毒蘑菇的信息。其描述如下：

> 该数据集描述了伞菌属和环柄菇属（第 500 ~ 525 页）的 23 种伞菌假定样本。每个品种都被定义为“确定可食用”“确定有毒”“可食性未知”或“不推荐食用”。最后一种定义可以视为有毒。该指南明确表示，关于蘑菇可食性的判断，并不存在像毒橡树和毒常春藤那样“三出复叶，勿食勿动”的简单规则。

每个数据点（实例）都代表一个蘑菇样本，每个样本都由其可观察到的属性（或称特征）进行描述。表 3-1 列举了 20 余个属性及属性值。每个实例中每个属性仅取一个值，如**菌褶颜色** = **黑**。我们选取了数据集中的 5644 个实例，其中包含了 2156 种毒蘑菇和 3488 种可食用蘑菇。

这是一个分类问题。这是因为存在目标变量**可食性**，而其取值为**是**（可食用）或**否**（有毒）两类。在训练数据集中，每一行的目标变量都有一个值。我们将用信息增益来回答“哪个属性能最好地区分蘑菇的可食性（**可食性** = **是**或**可食性** = **否**）”。这是一个基本的属性选择问题，在规模更大的问题中，可能需要从成百上千的属性中选出最有用的 10 个或 50 个属性。这么做是因为，对于某些数据挖掘问题来说这些属性数目太多，或其中无用的属性太多。为了简化，我们只选择一个而非 10 个最有用的属性。

注 4：详见加州大学欧文分校的机器学习仓库网页。

表3-1：蘑菇数据集的属性

属性名称	可能取的值
菌盖形状	钟形、圆锥形、凸形、平展形、圆球形、凹形
菌盖表面	纤维状、带凹槽、带鳞片、光滑
菌盖颜色	棕色、浅黄色、浅黄褐色、灰色、绿色、粉色、紫色、红色、白色、黄色
是否有斑点	有、无
气味	杏仁味、茴香味、木焦油味、鱼腥味、腐臭味、霉味、无味、刺鼻气味、辣味
菌褶与菌柄连接方式	直生、延生、离生、弯生
菌褶间隔	紧密、拥挤、较远
菌褶尺寸	宽、窄
菌褶颜色	黑色、褐色、浅黄色、巧克力色、灰色、绿色、橙色、粉色、紫色、红色、白色、黄色
菌柄形状	上细下粗、上粗下细
茎根	球状、棒状、杯状、等粗、根状、具根、无根
蘑菇圈以上的菌柄表面	纤维状、带鳞片、覆有丝状软毛、光滑
蘑菇圈以下的菌柄表面	纤维状、带鳞片、覆有丝状软毛、光滑
蘑菇圈以上的菌柄颜色	褐色、浅黄色、浅黄褐色、灰色、橙色、粉色、红色、白色、黄色
蘑菇圈以下的菌柄颜色	褐色、浅黄色、浅黄褐色、灰色、橙色、粉色、红色、白色、黄色
菌幕类型	内菌幕、外菌幕
菌幕颜色	褐色、橙色、白色、黄色
蘑菇圈数	无、1 个、2 个
蘑菇圈类型	蛛网型、隐失型、发光型、大型、无蘑菇圈、下垂型、外壳型、区域型
孢子印颜色	黑色、褐色、浅黄色、巧克力色、绿色、橙色、紫色、白色、黄色
居群	丰富的、群居的、许多的、散落的、较少的、独居的
栖息地	草丛、落叶、草地、小径、城市、荒地、森林
可食性（目标变量）	是、否

由于我们已经知道衡量信息增益的方法，所以目前的任务变得很直观：找出能产生最大信息增益的属性。

为此，我们需要先分别计算按每个属性进行分类后得到的信息增益。公式 3-2 中的信息增益由父集和一组子集来计算。对于每次计算而言，父集指的是整个数据集。首先需要计算的是**熵 (父集)**，即整个数据集的熵。如果目标变量的两个取值在数据集内完美地均匀分布，那么这个数据集熵值为 1。而由于目前的数据集存在轻微的不平衡（可食用蘑菇略多于毒蘑菇），所以其熵值为 0.96。

为了形象地展示熵值减小的过程，本书将使用一系列关于蘑菇分类的熵图（见图 3-6 至图 3-8）。由于可以根据不同的属性对整个数据集做出不同的划分，所以每张图仅从两个维度来描述该划分下的整个数据集的熵。x 轴表示当前数据集占整个数据集的比例（从 0 到 1），y 轴表示给定数据集的熵（同样从 0 到 1）。阴影区面积则表示根据不同属性划分后的（或未划分前的，见图 3-6）整个数据集的熵值。要找到最低熵值，就要使图中阴影区的总面积尽可能地小。

图 3-6 展现了整个数据集的熵。在此图中，熵值取理论上的最高值时，图中全部区域被阴影覆盖；熵值取理论上的最低值时，图中全部区域为空白。这种图可以很好地使数据在不同划分方式下的信息增益可视化，因为任何一种划分方式都能表现为图中的一组矩形区域：矩形的宽度代表划分后的子集在数据集中的比例，而其高度则代表子集的熵值。划分后的数据集的信息增益的加权和恰好为图中阴影区面积的总和。

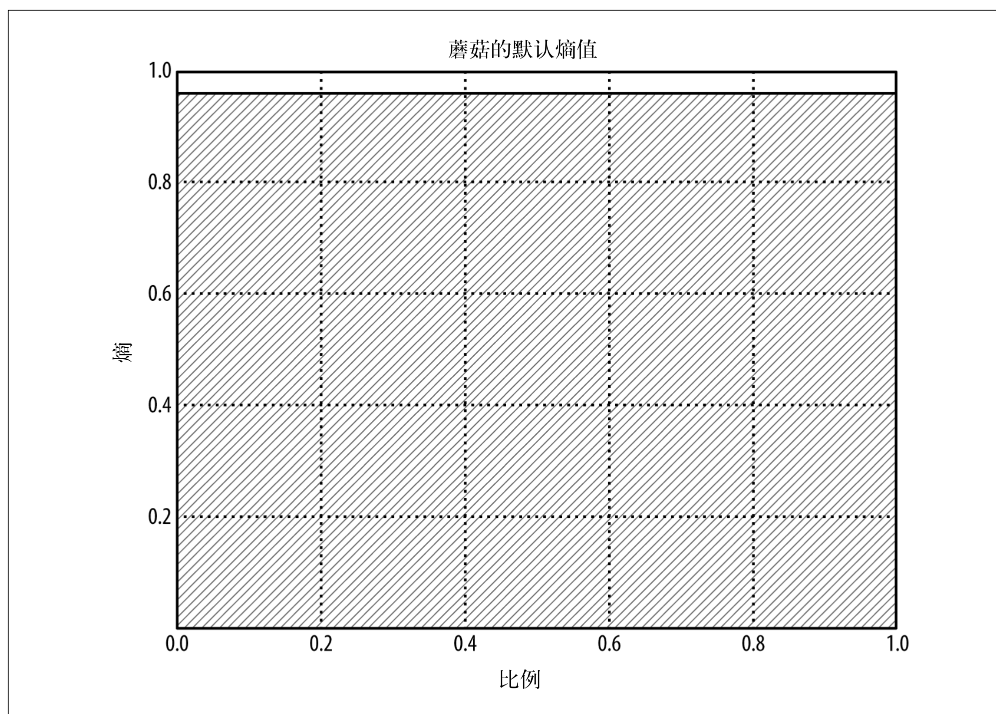


图 3-6：整个蘑菇数据集的熵图。因为整个数据集的熵是 0.96，所以有 96% 的区域是阴影区

因为整个数据集的总熵值是 0.96，所以在图 3-6 中，阴影区面积的上边界为横线 $y = 0.96$。我们可以把该值作为初始熵值，而任何用富信息属性导出的熵图中阴影区的面积都应该更小。下面，我们比较三个样本属性所对应的熵图。由于单个属性的不同取值在数据集中出现的频率不同，所以基于每个属性产生的数据划分方式也不同。

图 3-7 按菌褶颜色对数据集进行了划分，属性值包括 y（黄色）、u（紫色）和 n（褐色）等。每个值对应的矩形宽度代表了菌褶颜色为该值的数据点在整个数据集中所占比例，而其高度则为这个数据子集的熵值。可以看出，菌褶颜色这一属性降低了整体的熵值，因为图 3-7 中的阴影区面积明显小于图 3-6 中的阴影区面积。

类似地，图 3-8 展示了如何利用孢子印颜色这个属性降低信息的不确定性（熵值）。属性值中的一小部分，如 h（巧克力色），恰好可以对目标变量值进行明确的划分，因此产生了熵值为 0 的矩形阴影区。但请注意，这部分数据子集仅占整个数据集的 30%，并不是很多。

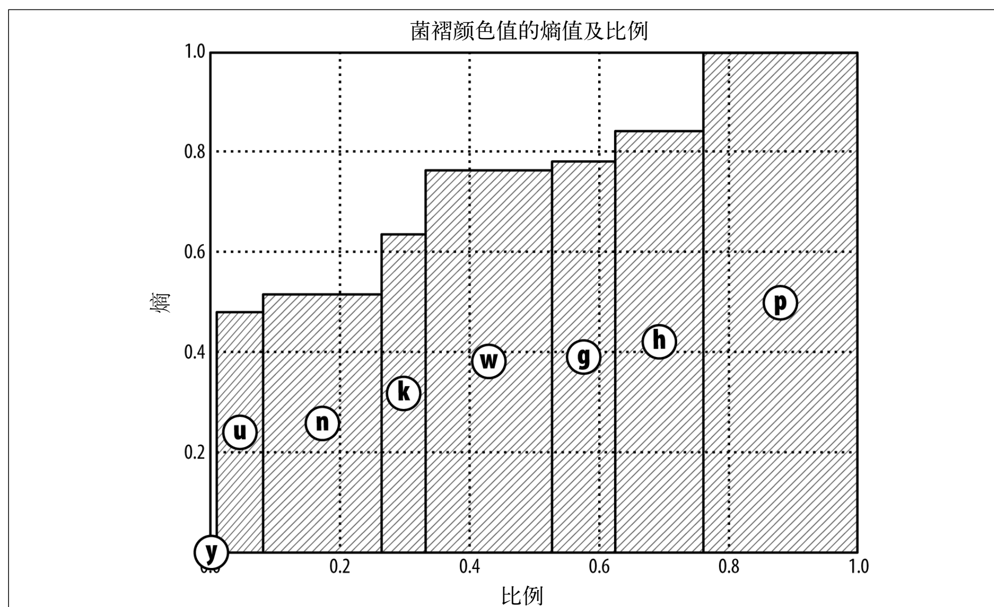


图 3-7：按菌褶颜色分类时的蘑菇数据集的熵图。阴影区面积相当于熵值的加权总和，每个矩形阴影区对应根据不同菌褶颜色划分出的子集，矩形阴影区的高度为该数据子集的熵，矩形阴影区的宽度则代表该数据子集在全体数据中所占的比例

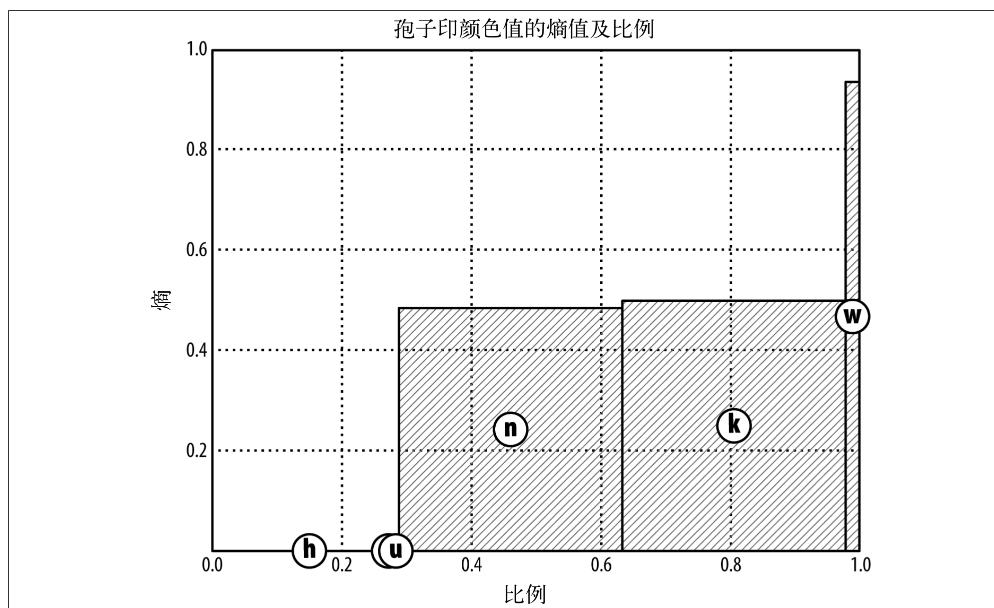


图 3-8：按孢子印颜色分类时的蘑菇数据集的熵图。阴影区面积相当于熵值的加权总和，每个矩形阴影区对应根据不同孢子印颜色所划分的数据子集，矩形阴影区的高度为该数据子集的熵，矩形阴影区的宽度则代表该数据子集在全体数据中所占的比例

图 3-9 按气味对数据集进行了划分。该属性的大部分值，如 a（杏仁味）、c（木焦油味）和 m（霉味），都产生了熵值为 0 的划分。仅有 n（无味）的熵值较大（约为 0.2）。实际上，气味这个属性在整个蘑菇数据集中具有最大的信息增益。它能够将数据集的整体熵值降到 0.1，因而其信息增益为 0.96 – 0.1 = 0.86。这意味着什么呢？许多种气味完全可以用于区分蘑菇是有毒的还是可食的，因而气味对于分辨蘑菇的可食性是一个非常好的富信息属性。[5] 如果你想仅仅根据一个特征来构建模型以判断蘑菇的可食性，那么气味就是最好的选择；如果你想建立一个更复杂的模型，那么最好也先从气味这个属性开始，随后再考虑加入其他属性。而这正是下一节的主题。

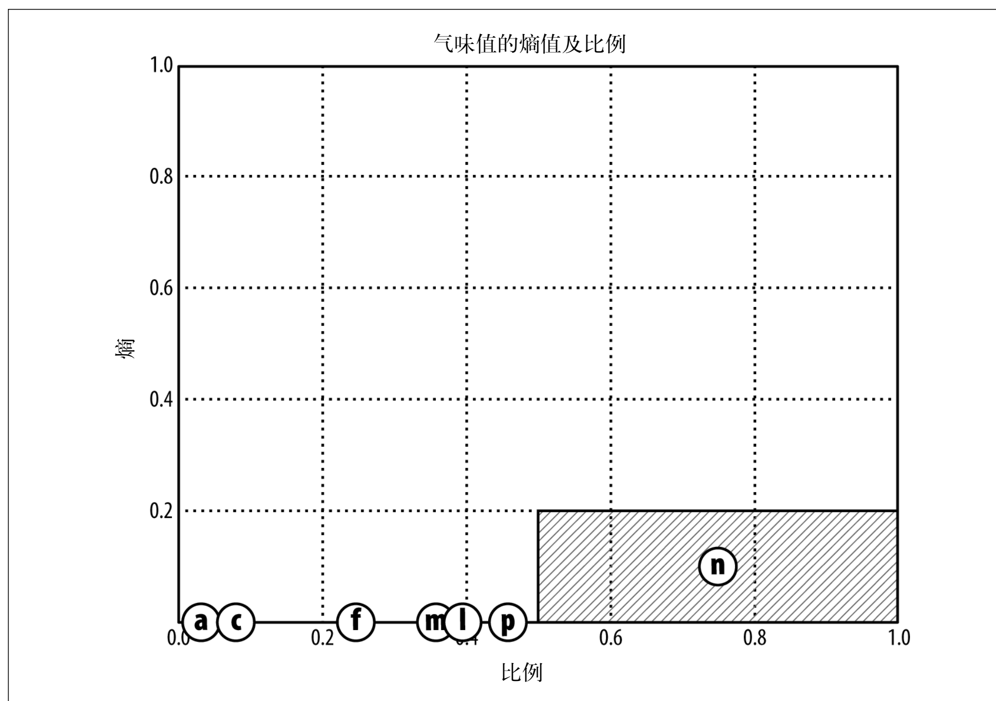


图 3-9：按气味分类时的蘑菇数据集的熵图。阴影区的面积相当于熵值的加权总和，每个矩形阴影区对应根据不同气味所划分的数据子集，矩形阴影区的高度为该数据子集的熵，矩形阴影区的宽度则代表该数据子集在全体数据中所占的比例

3.2.3 使用树形结构模型进行有监督的划分

本书已经介绍过数据科学的一个基本概念：从数据中选择富信息属性。接下来将继续讨论如何构建有监督的划分，因为选择属性虽然非常重要，但是单单进行这一步并不足以解决数据挖掘的问题。如果只选择出信息增益最大的一个变量，我们会得到一个非常简单的划

注 5：当然，这一切的前提是气味可被精确地测量。如果你的嗅觉较差，那么最好不要冒险。坦白地讲，你最好不要把自己的生命押在本书示例的数据挖掘结果上。当然，这并不妨碍我们把它作为一个不错的学习示例。

分方式；但如果要选择多个信息增益较大的属性，我们却还不清楚怎么把它们组合在一起。前文的例子中，我们尝试过使用多个属性进行数据划分，如“居住在纽约市的中年专业人士的平均用户流失率为 5%”。现在我们将巧妙地应用之前探讨过的关于选择重要属性的概念，来引出一个多变量（或多属性）的有监督的划分方法。

试想将数据的划分以树状的形式呈现。如图 3-10 所示，图中的树根在上，树冠朝下。这棵树由**节点**（包括内部节点和终端节点）和内部节点间的分支构成。树状图中的每个内部节点都会对某一个属性进行检验，并将该属性不同的值或者其值的不同区间分为不同的分支。从根节点沿分支向下看（即顺着箭头方向看），每条路径的末端都会有一个终端节点，称为**叶节点**。这棵树构建了一种数据划分方法，任意一个数据点在该树中对应并仅对应一条路径，也就是仅对应一个叶节点。换句话说，每个叶节点对应一个分组，而通向其的路径上的各个属性及其取值则给出了该分组的特征。因此，图 3-10 中最右边的路径所对应的分组为“年长、未就业、账户余额较多”。因为每个叶节点都包含了一个目标变量值，所以这棵树是**有监督**的划分。由于讨论的是分类问题，所以此处每个叶节点包含的是该分组的分类类别。我们称这样的树为**分类树**，或通俗点，称为**决策树**。

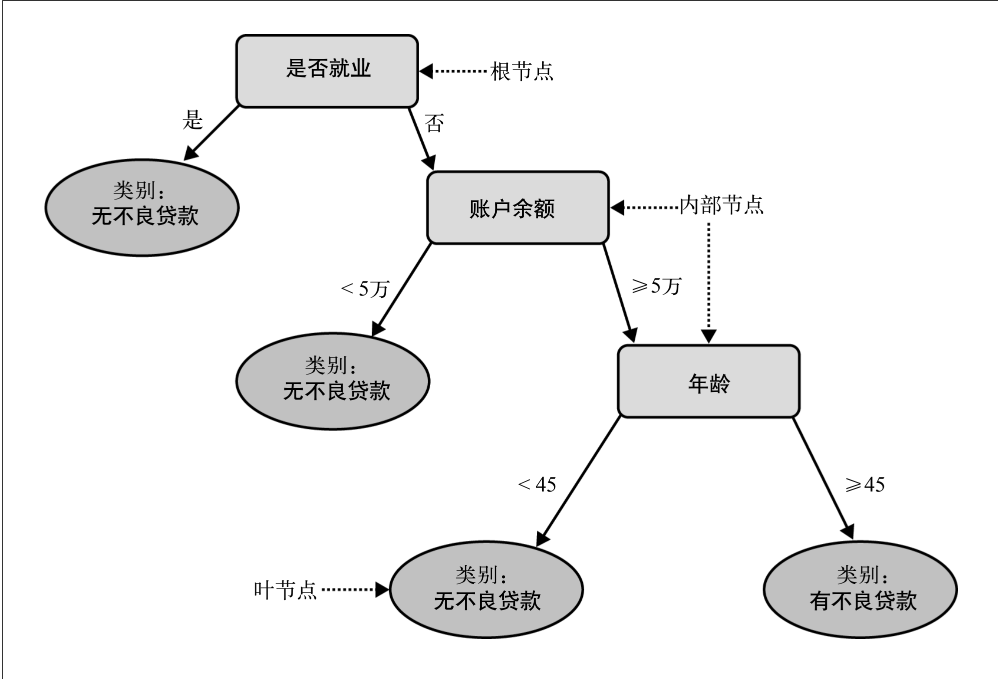


图 3-10：一个简单的分类树

分类树通常被用作预测模型，即“树形结构模型”。实际应用中，当拿到一个分类未知的实例时，我们可以寻找它对应的分组，并使用叶节点所对应的类别值来预测其类别。其实现方法是，从根节点开始，根据实例的具体属性的值来选择分支，向下遍历内部节点。树的非叶节点往往被称为“决策节点”，因为在向下遍历时，在每个节点上，都需要根据某

个属性的值选择向哪个分支遍历。沿着这些分支，最终会抵达终端节点。而根据其所对应的分类类别，就可以给出最终的分类预测结果。树中任意一个子节点有且仅有一个父节点，树中也不会出现闭环。树中的分支永远“指向下方”，故而每个实例最终都将到达某个叶节点，被赋予某个类别值。

思考一下，该如何使用图 3-10 中的分类树对图 3-1 中叫 Claudio 的人进行分类。Claudio 的属性分别为：**账户余额** = **11.5 万**、**是否就业** = 否、**年龄** = **40 岁**。从检验变量**是否就业**的根节点开始，其值为**否**，则选择右侧分支。下一个接受检验的变量是**账户余额**，其值为 11.5 万美元，大于 5 万美元，则选择右侧分支。再下一个变量是**年龄**，其值为 40，则选择左侧分支。最终我们来到**类别** = **无不良贷款**的叶节点，因此预测 Claudio 贷款不会违约。换句话说，我们把 Claudio 划分进了一个定义为**账户余额** = **11.5 万**、**是否就业** = 否、**年龄** < **45 岁**、分类为**无不良贷款**的分组中。

分类树是一种树形结构模型。后文中我们将看到，在实际商业应用中我们想预测的往往并非类别值本身，而是不同类别值的概率（如用户流失的概率或产生不良贷款的概率）。在此例中，**概率估计树**的叶节点就将包含这些概率，而不是仅仅给出一个简单的值。如果目标变量是数值型，相应地，**回归树**的叶节点包含的就是数值。无论如何，其基本概念是相通的。

树形图能生成一个符合期望的有监督的划分的模型，然而，尽管已经知道如何应用它来预测新实例的值，我们却不知道如何根据数据来构建这样的模型。现在我们来讨论这个问题。

有很多种技术都能从数据集中归纳出有监督的划分，其中最常见的一种就是构建树形结构模型（也即**树型归纳**）。这些技术之所以常见，是因为树形结构模型易于理解、归纳过程简洁（易于描述），而且易于使用。它可以稳定并相对高效地处理许多常见数据问题。行业中大部分数据挖掘工具包包含了某种树型归纳技术。

如何根据数据来构建分类树呢？综上所述，分类树的目的是进行有监督的划分，具体来说，就是根据每个实例的属性，将它们划分进目标变量值相近的子组中。我们期望每片“叶子”所对应的分组中包含的实例最好能属于同一类。

为了阐释分类树的归纳过程，我们再来看一下图 3-2 所示的简单示例。

树型归纳采取分而治之的方法，先从整个数据集开始，运用变量选择来找到产生尽可能“最纯”子集的属性。在本例中，给人分组的一种方法是基于身体形状：矩形或椭圆形。由此创建了如图 3-11 所示的两个组。这样的分组效果如何呢？矩形身体的一组在左侧，其中大部分是**有不良贷款者**，而仅有一个**无不良贷款者**，因此这组基本上是纯的。椭圆形身体组在右侧，大部分是无不良贷款者，但有两个**有不良贷款者**。不过这仅是对前文属性选择概念的一次直接应用。让我们暂且把这种分类方法视为产生信息增益最大的一种。

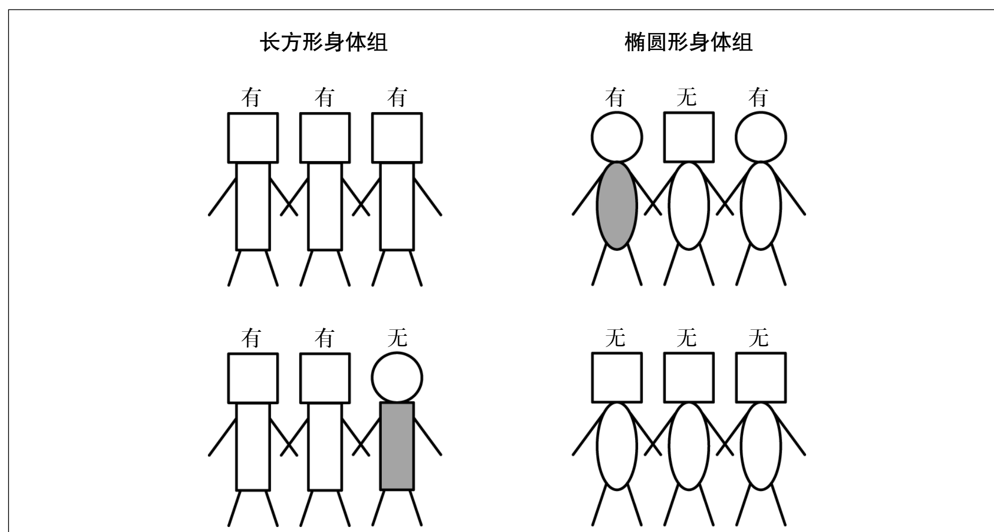


图 3-11：第一次划分，根据身体形状（矩形 / 椭圆形）进行划分

从图 3-11 中可以看到树型归纳的优美之处，以及其受欢迎的原因。左右两个子组只不过是最初要解决的问题的缩小版，只需**递归**地对每个子集应用属性选择，直至最终找出分组效果最佳的属性即可。因此在此例中，我们对椭圆形身体组进行递归式的处理（见图 3-12）。为了对这个组再次分组，我们使用另一个属性：脑袋形状。这样，该组又分为图中右侧所示的两组。这次分组效果又如何呢？每个新组都有单一的目标值：四个（方形脑袋组）**无不良贷款**和两个（圆形脑袋组）**有不良贷款**。因为这两组的类别标签都是"最纯"的，所以无须继续分组。

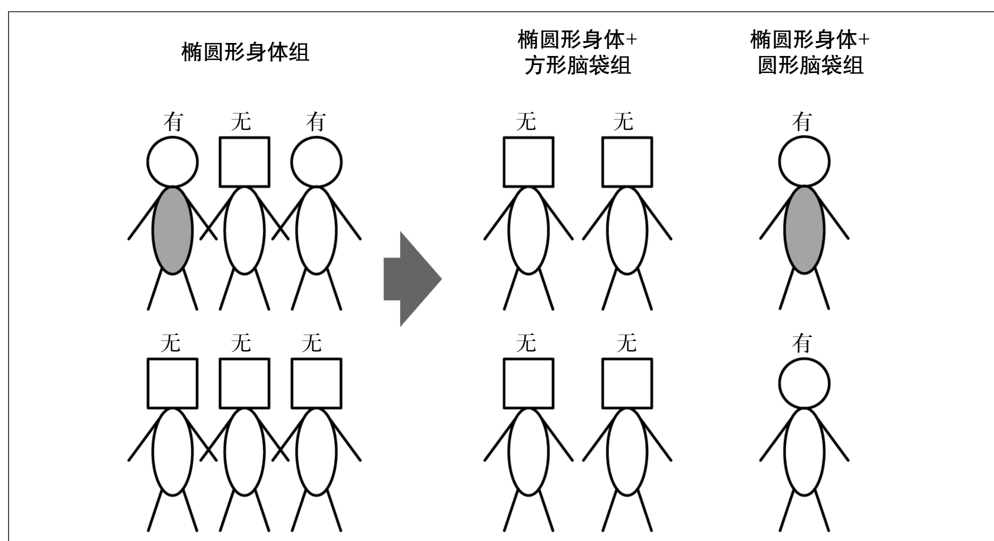


图 3-12：第二次划分，根据脑袋形状对椭圆形身体组进行再次划分

我们还未对图 3-11 中左侧的矩形身体组做任何处理，现在来看看它如何继续分组。组中含有五个**有不良贷款**和一个**无不良贷款**，有两个属性可作为分组依据——脑袋形状（方形 / 圆形）和身体颜色（白色 / 灰色），这二者都能起作用。此处随机地选择身体颜色作为分组依据，由此产生了如图 3-13 所示的分组。由于这些分组都是纯的（均只包含一种目标变量值），所以无须继续进行。所有这些分组对应的分类树见图 3-14。

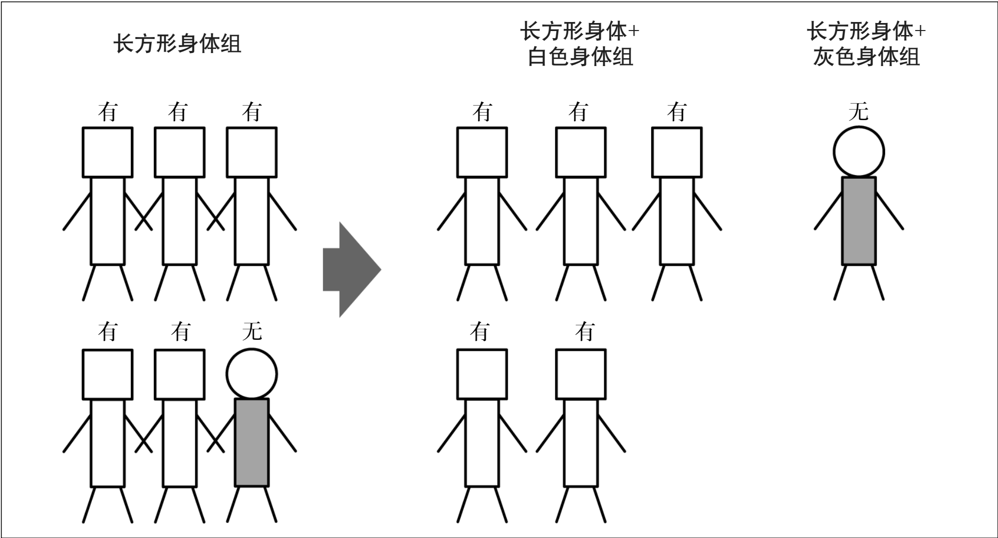


图 3-13：第三次划分，根据身体颜色对矩形身体组进行划分

总之，分类树型归纳的过程是一个递归地分而治之的过程，每一步的目标都是选出能把当前数据集分组成（对目标变量来说）尽可能纯的子集的属性。递归地进行这种划分，一步一步直到结束。在选择属性的时候，我们需要测试所有的变量，并选出能够使所分出的子集最纯的属性。那么何时结束呢？（换言之，何时停止递归？）显然，当节点是纯的，或所有变量均已被分组时，应该停止。但我们可能会需要提前结束，而这个问题将在第 5 章讨论。

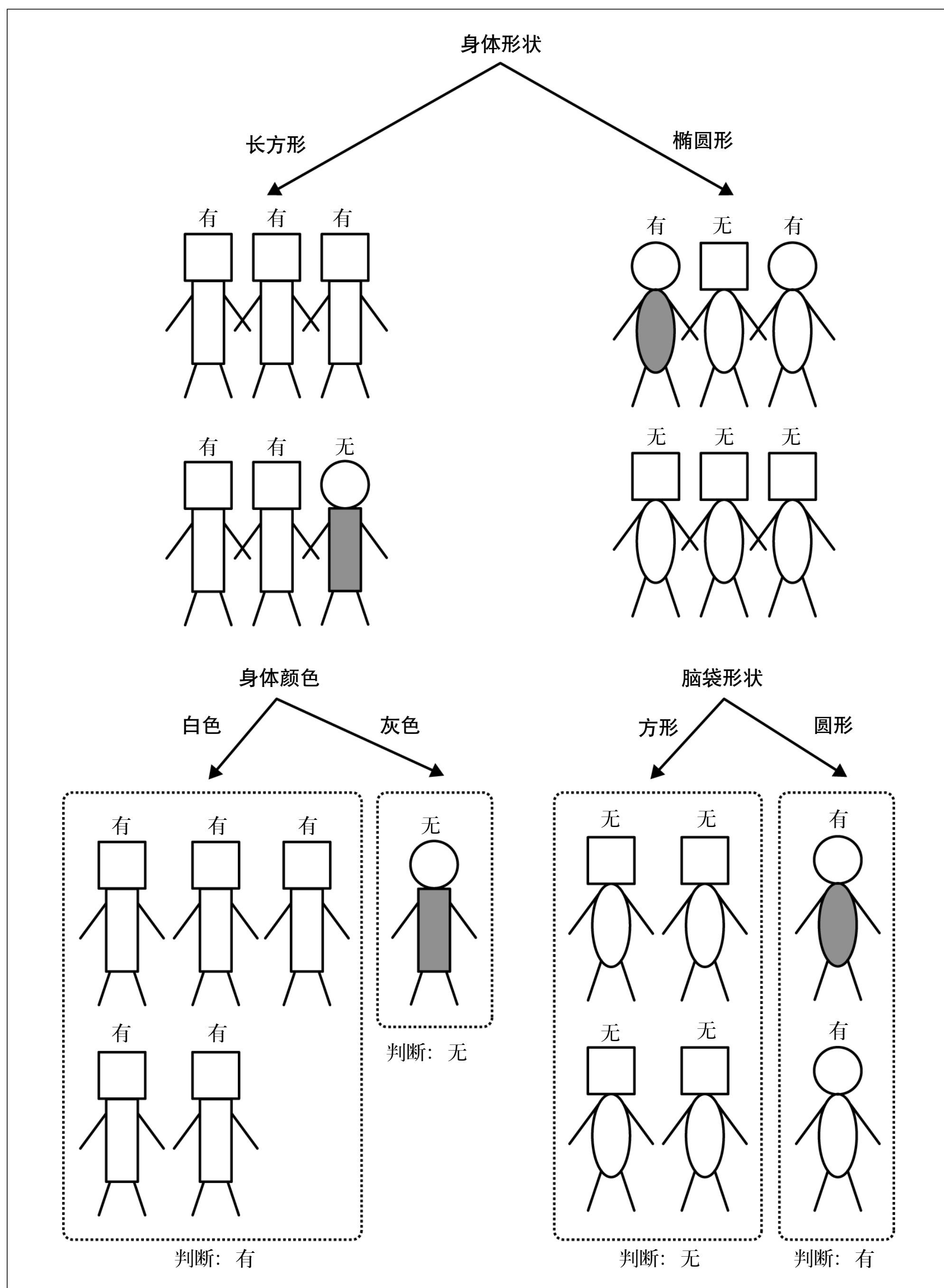


图 3-14：根据图 3-11 和图 3-13 的划分得到的分类树

3.3 划分的可视化

如果沿用把预测建模看作有监督的划分的说法，那么可视化地展现分类树是如何划分实例空间的会很有启发性。实例空间指的是由数据特征所描述的空间。一种常见的实例空间可视化的形式是描述某些成对特征的散点图。这些散点图将变量两两对比，来探索这些变量之间的关联和关系。

虽然数据可能包含十几个甚至百余个变量，但我们一次只能从两到三个维度对划分进行可视化。不过，仅从几个维度进行实例空间可视化，仍然有助于理解不同种类的模型，因为在这个过程中所得到的见解也同样适用于更高维度的空间。在比较差异较大的模型类别时，仅仅通过观察它们的形式（如一个数学公式相对于一组规则）或生成它们的算法可能不太容易做出比较。通常，更简单的方法是比较它们对实例空间的划分方式。

例如，图 3-15 展示了一个简单的分类树和其对应的二维实例空间图（*x* 轴代表账户余额，*y* 轴代表年龄）。分类树的根节点检验的是账户余额是否超过 5 万美元。与之对应的二维坐标图中"账户余额 = 5 万"的垂直虚线将平面划分成了"账户余额少于 5 万美元"和"账户余额不少于 5 万美元"两部分。左侧区域的实例账户余额不足 5 万美元，其中包含 13 个有不良贷款者（即"•"）和 2 个无不良贷款者（即"+"）。

根节点右侧分支指向账户余额不少于 5 万美元的实例。其下一个节点检验的是年龄是否超过 45 岁，与之对应的是二维坐标图中"年龄 = 45"的水平虚线，这条虚线仅出现在右半部分，因为这次划分仅针对账户余额超过 5 万美元的实例。该节点左侧分支指向的实例年龄决策节点是"年龄低于 45 岁"，与之对应的是二维坐标图的右下部分，代表"账户余额不少于 5 万美元且年龄低于 45 岁"。

注意，每个内部节点（决策节点）都对应对实例空间的一次划分，而每个叶节点都对应实例空间中一个未划分的区域（即总体的一个分组）。每当沿着一条路径离开某个决策节点后，我们都仅关注该次划分所产生的两个或多个子区域中的一个。随着向下遍历的过程，我们面对的实例空间子集也会越来越目标明确。



决策线与超平面

对实例空间进行划分的线被称作**决策线**（二维空间中），一般也被称作**决策平面**或**决策边界**。因为分类树中的内部节点是根据某个变量的取值进行检验的，所以这个节点所对应的决策边界总是垂直于该变量所在的坐标轴。在二维空间中，决策边界要么是水平的，要么是垂直的。若数据中有三个变量，则实例空间也是三维的，那么分类树的每个决策边界就是一个二维平面。在更高维的空间中，由于分类树的每个节点检验一个变量，而这可以视作确定了决策边界中的一个维度，所以，如果一个问题有 n 个变量，那么分类树的各个节点就可以在其实例空间中创建一个 $n-1$ 维的"超平面"决策边界。

在数据挖掘方面的文献资料中，**超平面**一词经常泛指一切分界平面，也就是说，不管什么平面都可以用它指代。别被这个术语吓到，你只要把它想成是泛指的线或平面的即可。

决策平面可能还有一些其他形式，后面会提到。

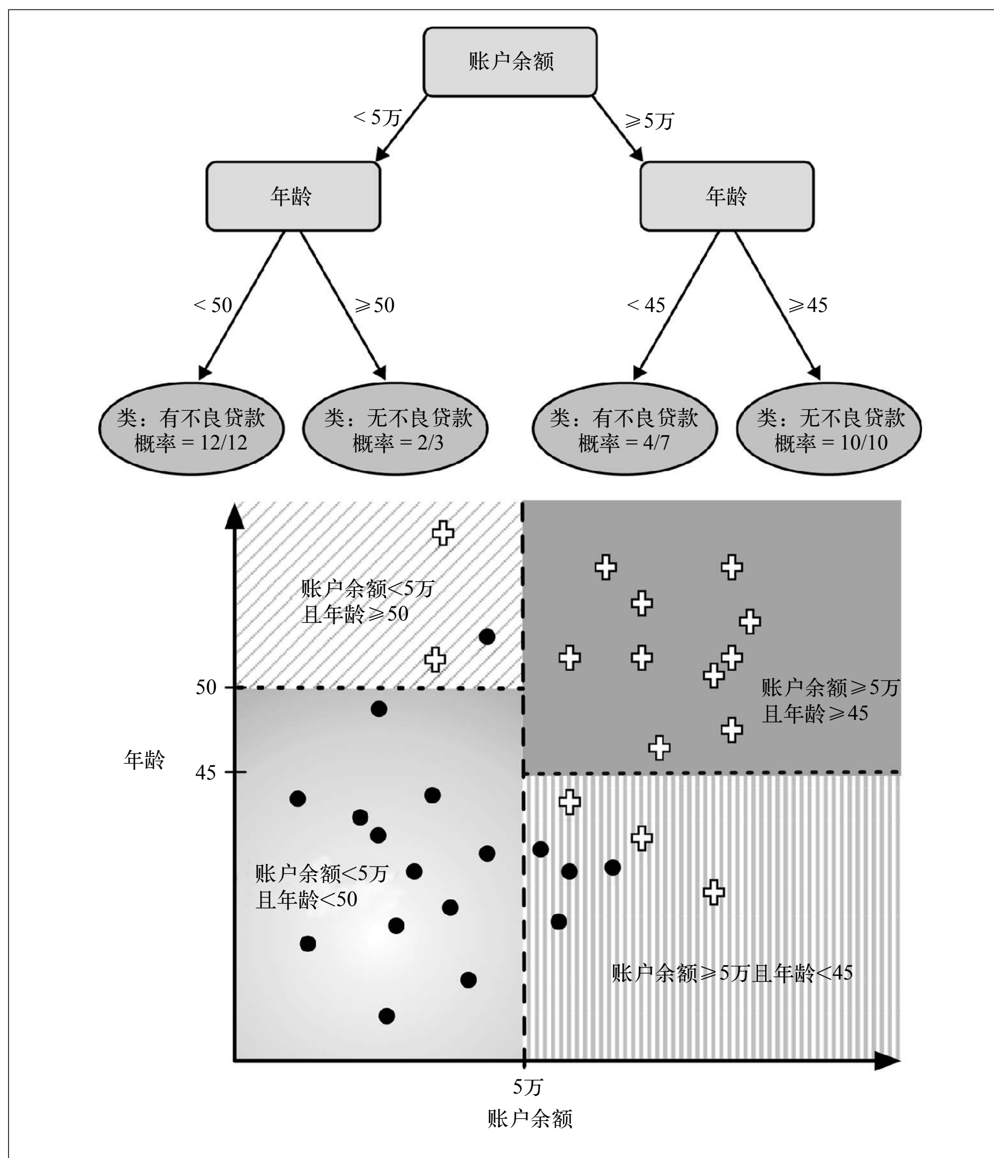


图 3-15：一个分类树及其所对应的实例空间的划分。“•”指有不良贷款，“+”指无不良贷款。不同的阴影区域代表了分类树的不同叶节点所对应的实例空间的划分

3.4 把树视作规则组

在结束对分类树的说明并开启下一个话题之前，有必要提一下分类树的另一种表现形式：逻辑声明。回想一下图 3-15 中的分类树，在对一个新出现的实例进行分类时，我们会从根节点开始，经过一系列属性检验后到达某个叶节点，最后得到该实例的类别的预测值。如

果从根节点沿一条单一路径抵达叶节点，并整合沿路径出现的所有条件，我们就能得到一条规则。每条规则都由沿路径进行的各属性检验的并集构成。例如，从根节点开始，如果始终选择左侧分支，将得到如下规则：

如果（账户余额少于 5 万美元）且（年龄低于 50 岁），那么分类为有不良贷款者。

根据树中每一条可能的路径，我们得到如下三条规则：

如果（账户余额少于 5 万美元）且（年龄不低于 50 岁），那么分类为无不良贷款者。
如果（账户余额不少于 5 万美元）且（年龄低于 45 岁），那么分类为有不良贷款者。
如果（账户余额不少于 5 万美元）且（年龄不低于 45 岁），那么分类为无不良贷款者。

分类树与上述的规则组是等价的。如果你觉得这些规则看上去有许多重复之处，那是因为它们确实是重复的：分类树就是一般规则中的条件聚集起来构成的。每个分类树都能以这种方式表示为一组规则。至于哪个更清晰易懂，这就见仁见智了。在上述的简单例子中，两种形式都非常易于理解。而当模型逐渐变得庞大的时候，两者的易理解性就会出现差异，不同的人对此也会有不同的偏好。

3.5 概率估计

在许多决策问题中，比起单纯的分类，我们更希望得到信息量更大的预测结果。比如在用户流失预测问题中，我们不仅预测了用户是否会在合约到期后 90 天内续约，而且估计了用户在该时间段内不再续约的概率。这样的估计用处多多，本书将在后面几章中详细讨论，在此仅做简要介绍：你可以按流失概率对用户进行排序，并将有限的激励预算分配到最可能流失的用户身上；或者，你可以把这些预算分配到一旦流失预期损失最大的用户身上，为此你同样需要流失概率的估计值。一旦有了这些概率估计值，你就可以将其应用到许多更加复杂的决策过程中，本书将在以后的章节中详细描述相关内容。

仅给出简单分类而非类别概率的模型，还存在另外一个更加不易察觉的问题。以信用贷款违约的预测问题为例，在一般情况下，基本上在我们进行信贷评估的总体的每个分组中，其成员产生不良贷款的概率都非常低——远小于 0.5。如果在这种情况下构建了一个模型，来对违约情况进行分类（即是否有不良贷款），那么就会出现所有分组的成员都不倾向于违约，即分类都相同（无不良贷款）的情况。打个比方，如果在一个构造简单的分类树中，每个叶节点都被标注为“无不良贷款”，这就会让数据挖掘新手非常沮丧：忙了半天，结果竟然是没人可能会违约？不过，这**并不**意味着该模型毫无用处。不同分组产生不良贷款的概率可能的确大相径庭——只不过它们都小于 0.5 而已。如果根据这些信贷违约概率来进行信贷评估，就能大大降低风险。

因此，在有监督的划分中，我们希望每个分组（即分类树的每个叶节点）都给出不同类别下成员概率的估计值。图 3-15 基于不良贷款预测示例展示了一个常规的“概率估计树”模型，它不仅预测了类别值，还预测了类概率估计值。[6]

注 6：通常我们处理的是二元分类问题，如有不良贷款与否、用户流失与否。在这些情况下，往往只计算其中一类的概率 $p(c)$，这是因为另一类自然是 $1 - p(c)$。

幸运的是，目前我们讨论的树型归纳概念能够非常容易地导出概率估计树，而不仅是简单的分类树。[7] 前文中提过，树型归纳能将实例空间划分成类别尽可能纯（即低熵值）的区域。如果认同某一叶节点所对应的分组内的各成员类概率相同这一假设，那么我们便可以用每个叶节点中的实例数来计算某个类概率的估计值。比如，如果一个叶节点包含 n 个正实例和 m 个负实例，那么新实例为正的概率就是 $n/(n+m)$。这种方法被称为**基于频率**的类成员概率估计。

此时你大概会发现一个问题：当使用上述方法估计类成员概率时，我们对实例数极少的分组中的类成员概率估计可能会过度乐观。极端情况下，假设一个叶节点只含有一个类别碰巧为某个值的实例，我们可以说被划入这个叶节点的任意一个新实例属于该类别的概率是 100% 吗？

这种现象是数据科学中一个基本概念（**"过拟合"**）的一个示例，之后本书将用整整一章来论述它。为了本书结构的完整性，在此先简要说明一个简单方法，来解决在小样本情况下基于分类树的类概率估计而存在的过拟合问题。这时我们通常不会单纯地计算频率，而会用一种"平滑"后的基于频率的估计，称作"拉普拉斯修正"。其目的是减轻仅有几个实例的叶节点对类概率估计的影响。二元分类问题的类概率估计公式就变成了如下形式：

$$p(c) = \frac{n+1}{n+m+2}$$

其中 n 是叶节点中属于类 c 的实例数，m 是不属于类 c 的实例数。

在此用一个示例来比较使用和不使用拉普拉斯修正这两种情景。一个有 2 个正实例、没有负实例的叶节点，与另一个有 20 个正实例、没有负实例的叶节点基于频率的概率估计值（即 $p = 1$）相同。然而，前者的实例数太少，其估计值可能由极端情况导致，因此该估计需要调整。拉普拉斯公式将该估计平滑到了 $p = 0.75$，确实体现了它的不确定性。但拉普拉斯修正对于实例数为 20 的叶节点而言，影响就小多了（$p \approx 0.95$）。随着实例个数的增加，拉普拉斯修正的结果逐渐趋近于基于频率的估计值。图 3-16 展示了当实例数逐渐增加时，拉普拉斯修正对不同类别比例（2/3、4/5 和 1/1）的修正效果。图中对应的每个类别比例的水平实线代表未修正的（常数）估计值，相应的虚线则代表应用拉普拉斯修正后的估计值，前者是后者在实例数趋向无穷时的渐近线。

注 7：即使决策者用的是概率估计而非简单分类，它们也仍被称为分类树。

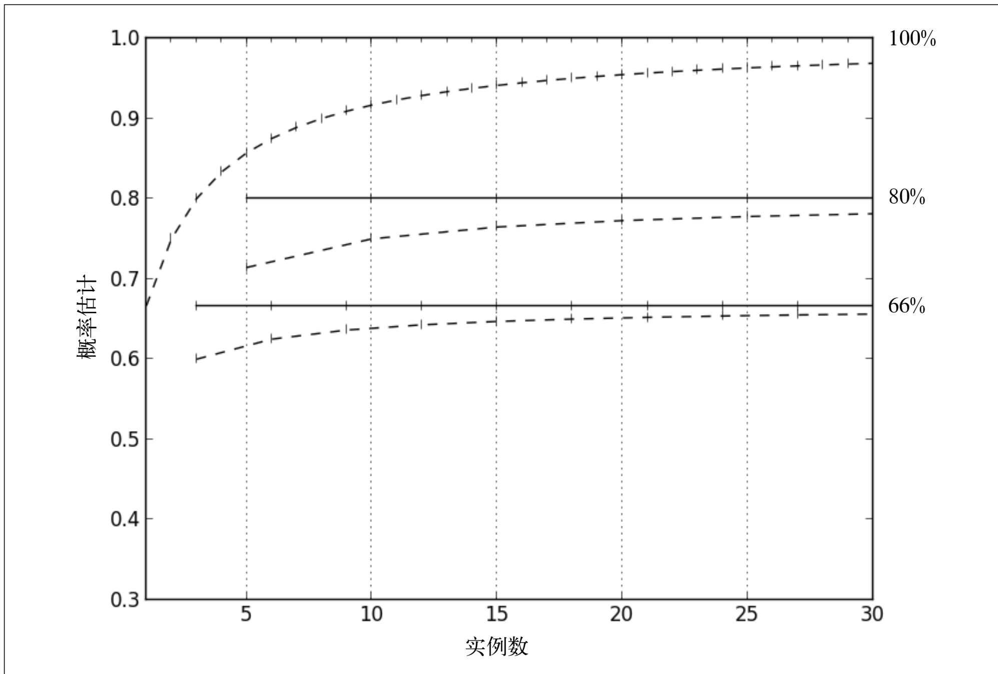


图 3-16：拉普拉斯修正对不同实例比率的平滑效果

3.6 示例：用树型归纳解决用户流失问题

在学完预测建模的数据挖掘基础技术之后，我们继续考虑用户流失问题。如何运用树型归纳来解决这个问题呢？

假设我们的历史数据集包含 20 000 个用户，在收集数据期间，这些用户要么续约，要么不再续约（即流失）。表 3-2 列出了描述用户所用的变量。

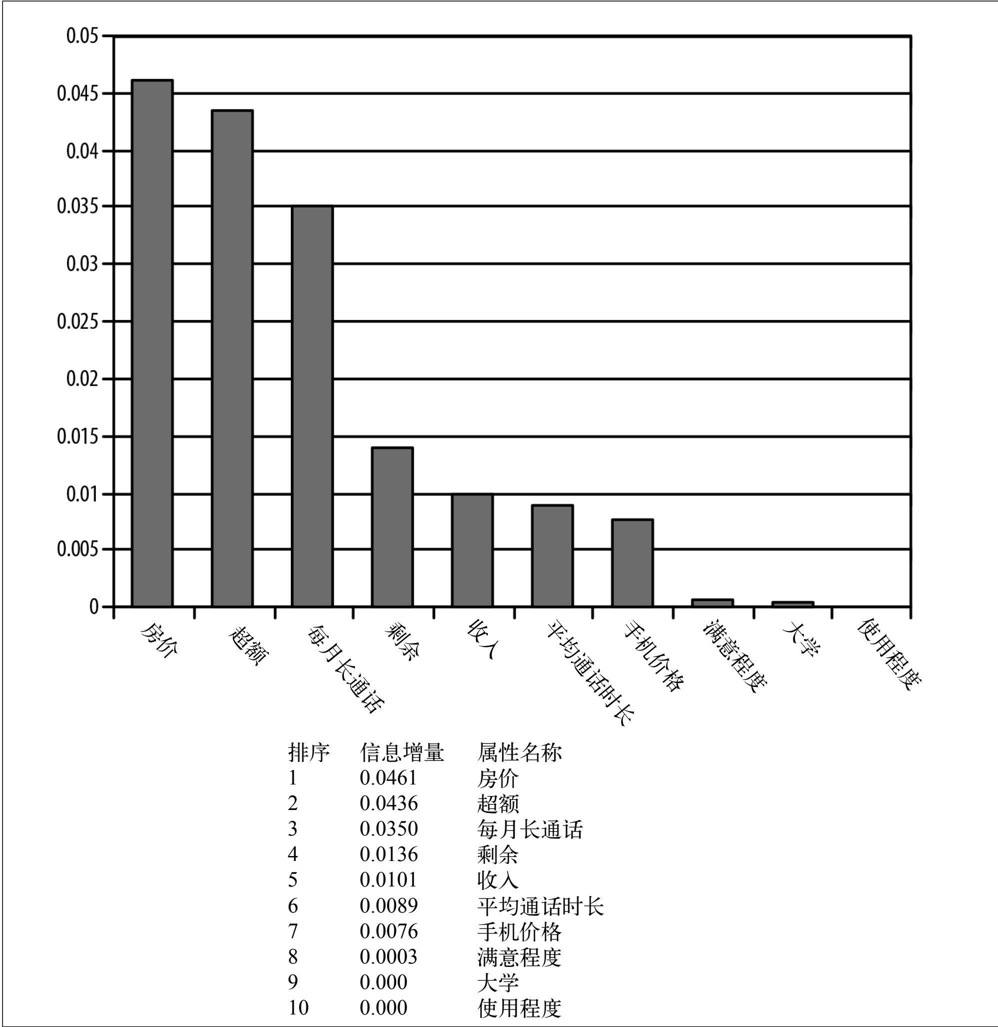
表3-2：通信公司用户流失问题中的用户属性

变　　量	解　　释
大学	该客户是否有大学学历？
收入	年收入
超额	月平均超额使用费用
剩余	月平均剩余分钟数
房价	房价估计值（根据人口区域普查）
手机价格	手机价格
每月长通话	月平均长通话数（不少于 15 分钟）
平均通话时长	平均通话时长
满意程度	满意程度
使用程度	用户自我评定的使用程度
流失（目标变量）	用户是否还留在公司（是否流失）？

这些变量所包含的基础人口统计信息和使用信息可从用户的应用和账户中取得。我们将根据这些数据，运用树型归纳技术来预测哪些新用户将会流失。

在根据以上变量构建分类树之前，最好知道**这些变量各自独立的预测效果如何**。为此，我们按照前文所述的方法测量了每个属性所产生的信息增益。尤其是，在整个实例集合中，我们将公式 3-2 分别应用于每一个变量，并计算它们所产生的信息增益。

结果如图 3-17 中的列表所示。可以看出，前三个变量——“房价”“超额”和“每月长通话”——的信息增益比其他变量高。[8] 出乎意料的是，“使用程度”和“满意程度”两者本身对用户流失的预测效果都不好。



排序	信息增量	属性名称
1	0.0461	房价
2	0.0436	超额
3	0.0350	每月长通话
4	0.0136	剩余
5	0.0101	收入
6	0.0089	平均通话时长
7	0.0076	手机价格
8	0.0003	满意程度
9	0.000	大学
10	0.000	使用程度

图 3-17：表 3-2 中的用户流失数据中的变量，按信息增益排序

注 8：注意，用户流失数据集中变量的信息增益比先前蘑菇数据集中变量的信息增益小很多。

将分类树算法应用于数据后，就得到图 3-18 中的分类树。图 3-17 表明，信息增益最高的变量“房价”位于树的根节点。这是符合预期的，因为这个变量往往被最先选择。第二好的特征，“超额”，也在树的上部。然而，树中变量的选择顺序与图 3-17 中的顺序并不完全相同，这是为什么呢？

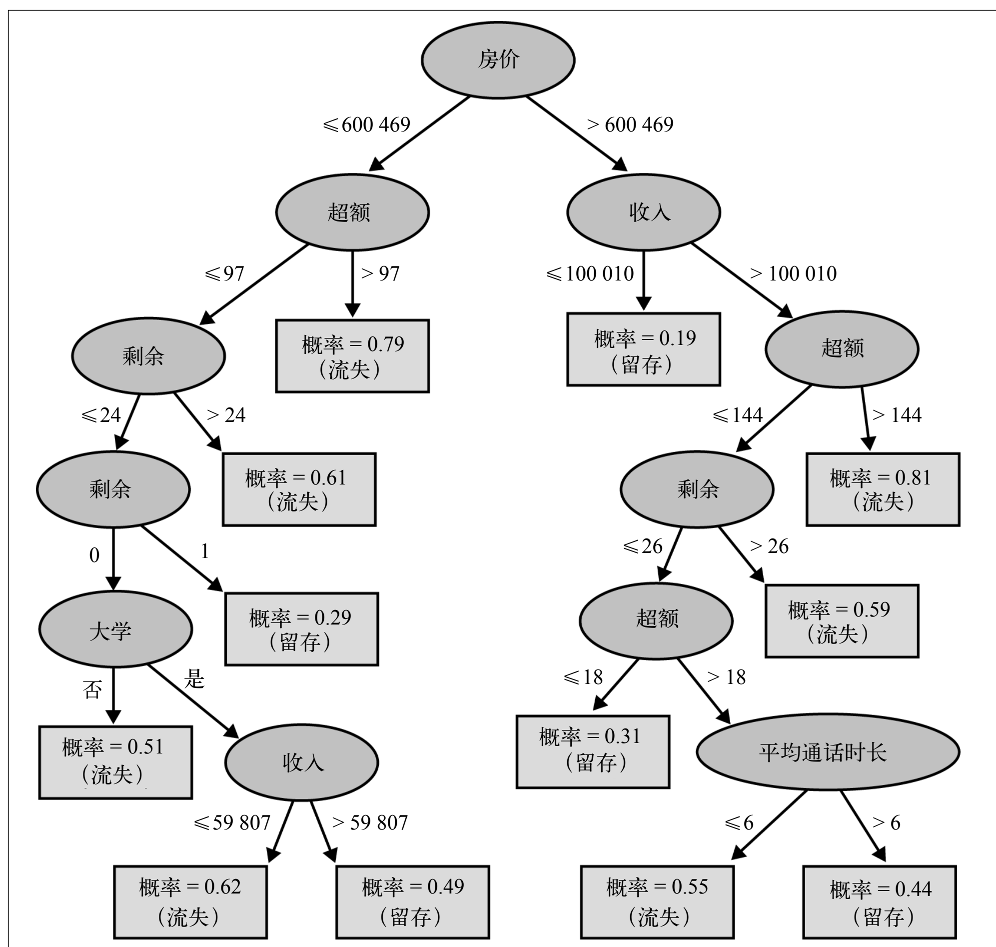


图 3-18：手机用户流失数据所构建的分类树。矩形的叶节点代表总体的分组，该分组由从根节点向下延伸出的路径定义。叶节点上的概率值是该分组下用户流失概率的估计值；括号内则是根据概率是否大于 0.5 来进行的分类决策 [比如：该划分下的个体是倾向于流失（CHURN），还是留存（STAY）？]

原因是，在图 3-17 的列表中，我们基于整个实例总体分别测量各变量的信息增益，并按照变量**独立的**表现给它们排序。而分类树中的节点则取决于其上一层的实例集合。因此除了根节点外，分类树中的特征的信息增益都不是基于整个实例集合来测量的。由于任意一个特征的信息增益都依赖于它所基于的实例集合，所以某些内部节点的特征的排序就可能与全局排序不同。

我们还未讨论如何决定终止分类树的分支。示例使用的数据集中含有 20 000 个数据点，而分类树的叶节点显然没有这么多。我们能一直持续地选择属性来划分数据，直到无数据可分为止吗？这样虽然也可以，但如此一来模型就会变得非常复杂，因此实际上我们需要早早停止。而这个问题与模型的通用性和过拟合密切相关。第 5 章会探讨过拟合。

思考一下该数据集的最后一个问题。在基于数据建立树形结构模型后，我们通过测量其准确率来衡量该模型的优劣程度。具体做法是：选取一个训练集，使其中的流失用户与未流失用户各占一半；用该训练集构建一个分类树，再将其应用于原数据集，看看有多少实例被正确地分类。最终，这个分类树分类的正确率是 73%。这引出了如下问题。

(1) 首先，你相信这个数字吗？如果把这个分类树应用于另一个源于相同数据集的 20 000 人样本，其精确度仍会是 73% 吗？

(2) 其次，即使你**的确**相信这个数字，可它真的意味着模型优良吗？换句话说，这个准确率为 73% 的模型能用吗？

第 7 章和第 8 章会回顾这些问题，并深入研究模型评估问题。

3.7 小结

本章介绍了预测建模的基本概念。预测建模是数据科学的主要任务之一，它通过建立模型来估计新个体的目标变量值。其间本章引入了数据科学的一个基本概念：找出并选择富信息属性。选择富信息属性本身也是一个有用的数据挖掘过程。面对一个庞大的数据集，我们现在能够找出其中的某些变量，它们要么能给出有关我们所关心的其他变量的信息，要么与该变量相关。比如，如果收集了在合约到期后短期内续约或不续约（即流失）的用户的历史数据，那么通过属性选择就可以找到人口统计方面或账户方面的变量，进而可以使用其中的信息来反映用户流失的可能性。衡量属性信息量的基本指标是信息增益，它基于一个被称作**熵**的纯度指标，另一个指标则是方差缩减。

富信息属性选择是常用的建模技术之一——树型归纳——的基础。树型归纳能够递归地找出数据子集中的富信息属性，同时把其实例空间划分为相似的区域。这样的划分之所以被称为“有监督的”，是因为它所尝试找出的分组，可以为要预测的量（即目标变量）提供越来越精确的信息。最终的树形结构模型将实例空间划分成一系列分组，而每个分组对应的目标变量预测值都不同。比如，如果目标变量的分类是二元的（如是否流失，或是否有不良贷款），那么分类树的每个叶节点就对应着总体中的一个分组，而各分组对应的类成员概率估计值各不相同。



作为练习，思考一下：如果用回归构建一个树形结构模型，它跟分类树有何不同？在你学过的分类树型归纳过程中，需要对哪些因素做出改变才可以生成回归树？

历史上，树型归纳由于具有通俗易懂、易于实施和计算廉价的优势，一直是一种非常受欢迎的数据挖掘方法。对树型归纳的研究至少要追溯到 20 世纪五六十年代。最早的树型归纳系统，包括 CHAID（卡方自动交互侦测器，Kass, 1980）和 CART（分类与回归树，

Breiman, Freidman, Olshen & Stone, 1984），至今仍被广泛应用。C4.5 和 C5.0 作为同样流行的树型归纳算法，其世系显而易见（Quinlan, 1986, 1993）。J48 则是 Weka[9] 包中对 C4.5 的重新实现（Witten & Frank, 2000; Hall 等 , 2001）。

实践中，在我们能从特定数据集中提取出的模型中，树形结构尽管不是精度最高的，效果却出类拔萃。在很多情况下，尤其是在应用数据挖掘的早期，使模型易于理解、便于解释是十分重要的。这一点不仅对数据科学团队本身十分有用，而且在他们和不懂数据挖掘的企业利益相关者交流成果时也是非常有用的。

注 9：一种数据挖掘软件。——译者注

第 4 章
用模型拟合数据

基本概念：基于数据寻找“最优”模型参数；选择数据挖掘目标；目标函数；损失函数

示例方法：线性回归；逻辑回归；支持向量机

我们已经了解到，预测建模就是根据其他描述性属性找出目标变量的模型的过程。在第 3 章，我们通过在逐步精确的数据子集中（或从几何角度讲，从逐步精确的实例空间的子空间中）递归地寻找富信息变量，构建了一个有监督的划分的模型。根据数据，我们不仅构建了模型的结构（即由树型归纳得到的树形模型），还得到了模型的数值型“参数”（即叶节点上的概率估计）。

另一种从数据集中提取预测模型的方法，是先确定模型的结构，而使模型的数值型参数待定。然后再通过数据挖掘，根据特定的训练数据集计算出最佳参数值。常见的情形是，模型由含有一系列数值变量的参数化的数学函数或公式构成，而我们既可以基于领域知识，从理论上判断哪些变量对目标变量具有较好的预测性，也可以基于其他数据挖掘技术（如第 3 章介绍的属性选择方法）来决定模型需要使用哪些变量。数据挖掘系统中，模型的形式及其所用变量是确定的，数据挖掘的目的则是通过调整参数来使模型尽可能地拟合数据。这种一般方法被称作**参数学习**或**参数化建模**。



在统计学和计量经济学的某些领域中，“模型”是指未明确参数的模型。我们需要澄清，这仅是一个模型的结构，它在参数待定的情况下是无法使用的。

这个一般框架包含许多数据挖掘过程，而这些数据挖掘过程均基于**线性**模型，本书将展示其中一些最常用的。如果你学过统计学，那么你可能已经了解了一种线性建模技术：线性

回归。其中会有一些区别和我们已经学过的模型的区别相同，比如分类任务、类概率估计任务和回归任务之中的区别。在举例说明的部分，本书会展示一些常用技术。它们可以用来预测（或估计）未知数值、未知二元值（如某份文件或某个网页是否符合查询要求），以及事件发生的概率（如信贷欺诈、响应优惠活动、账户欺诈等）。

本章还会明确地探讨第 3 章中所绕开的话题：模型拟合数据的效果“好”究竟是什么意思？而这正是本章基本概念的关键——通过寻找“最优”模型参数用模型拟合数据——这也是在后续章节中将会继续出现的概念。正是由于这个概念的基础性，本章中的数学知识会相对较多。但本章仍会尽可能减少数学概念，以便数学基础较薄弱的读者放心阅读。

本章中的简化假设

本章主要是介绍和讲解参数化建模，为了突出讨论重点、避免过多使用脚注，在此做一些简化假设。

首先，为了方便讨论分类问题和类概率估计问题，本章将只考虑二元分类，即模型预测的事件要么发生，要么不发生。比如是否对优惠活动做出响应、是否离开公司或是否被欺诈等。尽管这些方法都可以推广到多元（非二元）分类中，但这只会增加不必要的复杂性。其次，因为本章主要是跟公式打交道，所以本章将假设其中所有属性都是数值型的。在需要使用这些公式时，有很多技术都可以将类别型（符号）属性转化为数值。最后，本章将忽略数值变量的尺度归一化问题。由于不同属性（如年龄和收入）的取值范围大不相同，因而需要被归一为统一的尺度，这样不仅可以提高模型的可解释性，还会带来一些其他的益处（本章将在后文探讨）。

在本章中，我们姑且忽略上述的复杂问题。然而，其实无论对于哪种数据挖掘技术而言，处理这些问题都是非常重要也非常必要的。

4.1 根据数学函数分类

第 3 章中讲到，树形模型可以表示为实例空间中的空间划分。如图 4-1 所示，其中的实例空间被水平或垂直的决策边界划分成了类似的区域。每个区域中所包含的实例都应该有相似的目标变量值。上一章中，我们还学会了如何使用熵这个指标来测量同质性，并以此来选择决策边界。

创建同质区域的主要目的，就是通过判断一个新的、从未出现过的实例会落入哪个区域来预测它的目标变量值。比如，如果一个新用户被划分入图 4-1 中的左下角区域，那么就可以说，该用户的目标变量值很可能是“•”；同理，如果它落入右上角区域，那么其目标变量值很可能是“+”。

实例空间这一视角非常有用：如果去掉与坐标轴平行的决策边界（见图 4-2），就能很清楚地看到更好的划分实例空间的方式。比如，如果能画一条斜线（如图 4-3 所示）——而不是任何一条坐标轴的垂线——作为决策边界，就能近乎完美地按类别对这些实例进行划分。

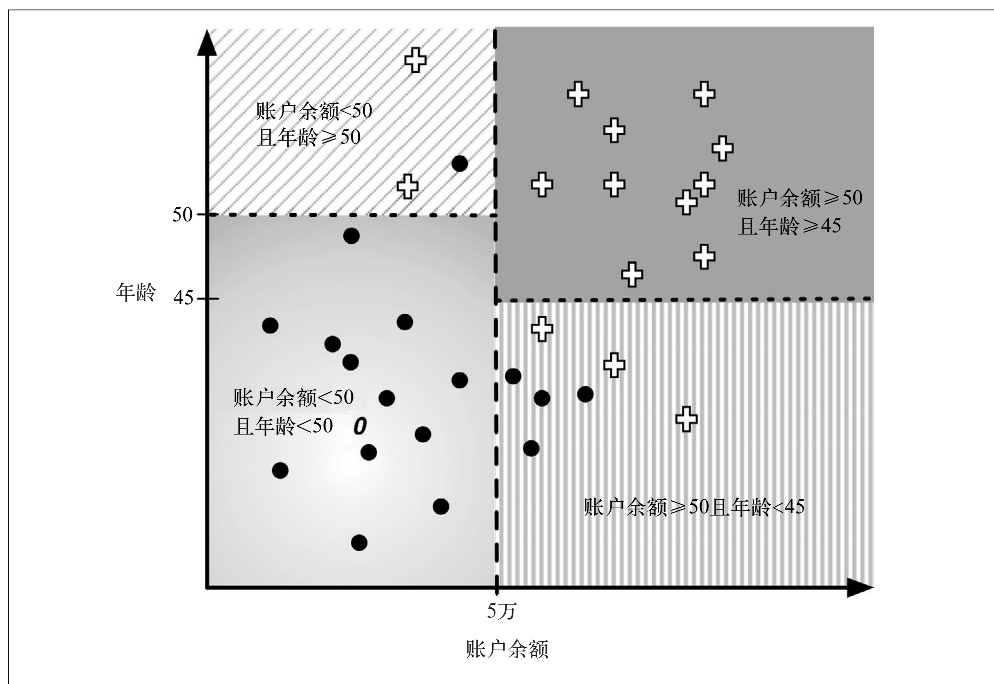


图 4-1：一个被分类树划分的数据集，包含四个叶节点

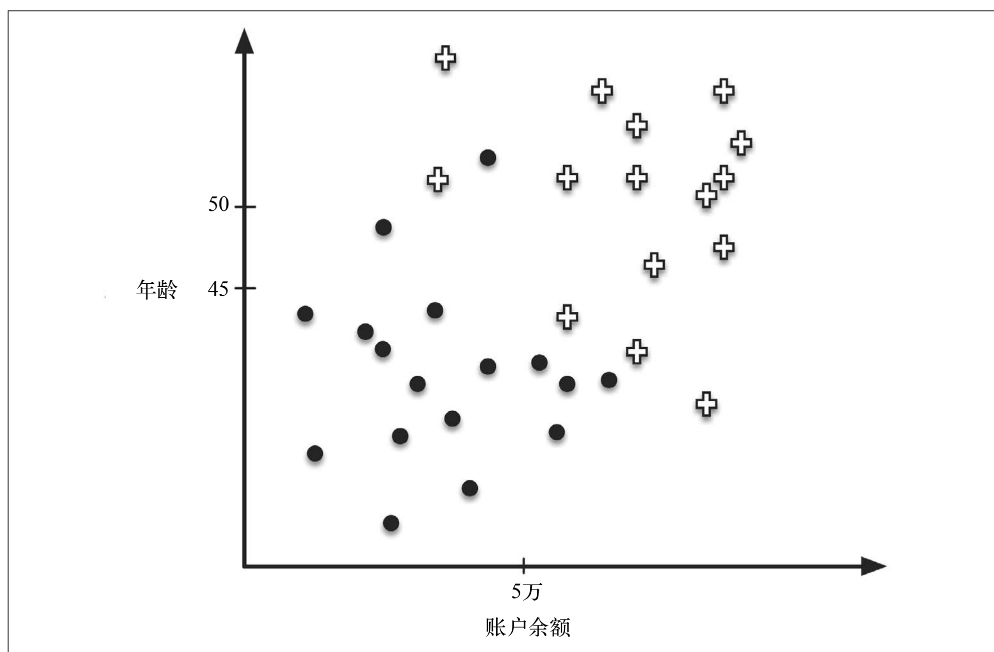


图 4-2：图 4-1 中的原始数据点，无决策线

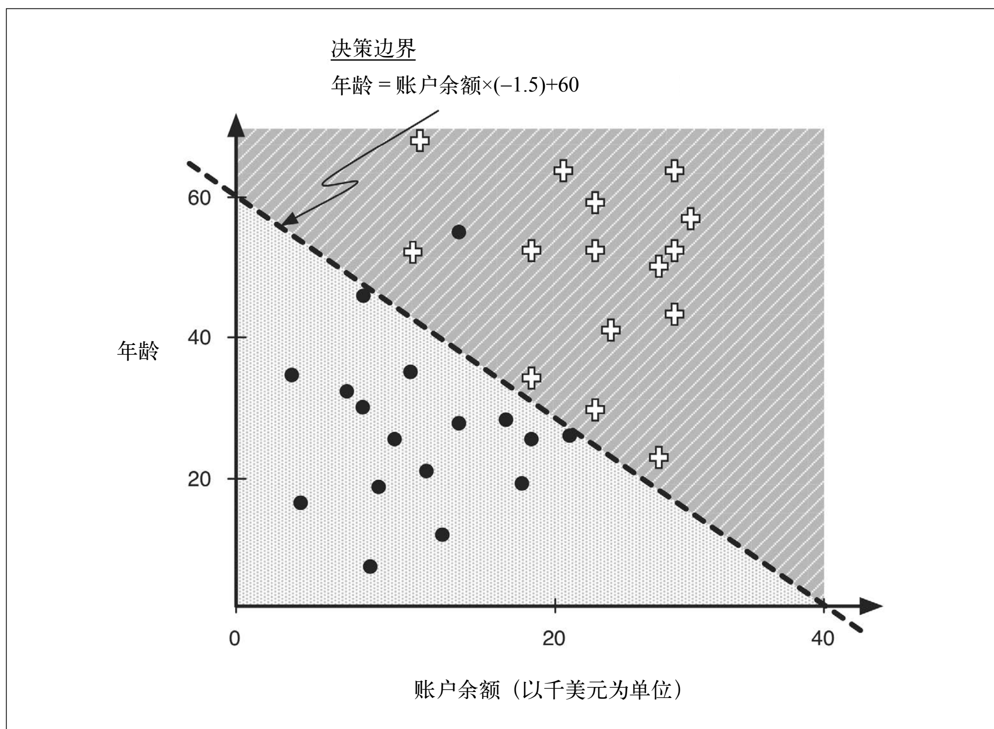


图 4-3：图 4-2 中的数据集用一条直线进行划分

这种方法被称作**线性分类器**，其本质上是多个属性值的加权和。本章将在后文中探讨它。

4.1.1 线性判别函数

我们的目标是用模型拟合数据，这时数学语言就能派上用场了。你应该学过，在二维坐标系中，直线的公式为 $y = mx + b$，其中 m 是斜率，b 是 y 轴的截距（即当 $x = 0$ 时 y 的值）。图 4-3 中的直线就可以用这种方式描述（账户余额以千美元为单位）：

$$\text{年龄} = (-1.5) \times \text{账户余额} + 60$$

我们把位于直线上方的实例 $\boldsymbol{x}$ 分类为“+”，而把直线下方的实例分类为“•”。将这个过程用数学语言重新组织一下，就得出了一条函数式，而该式即为本章所要探讨的所有技术的基础。该例中的决策边界的分类解析式可见公式 4-1。

公式 4-1：分类函数

$$\text{类别}(\boldsymbol{x}) = \begin{cases} + & \text{若} -1.0 \times \text{年龄} - 1.5 \times \text{账户余额} + 60 > 0 \\ \bullet & \text{若} -1.0 \times \text{年龄} - 1.5 \times \text{账户余额} + 60 \leqslant 0 \end{cases}$$

上式被称作**线性判别式**，因为该式能够判别分类，而决策边界的函数又是属性的线性组合（加权和）。在本例的二维空间中，线性组合对应一条直线；在三维空间中，决策边界则是

一个平面；而在更高维度的空间中，它会是一个**超平面**（见 3.3 节）。对我们来说最重要的是模型可以用属性值的加权和来描述。

因此，该线性模型是多变量有监督的划分的一种不同形式。进行有监督的划分的目的，仍旧是把数据划分成目标变量值不同的区域。不同的是，实现该多属性建模的方法是构建一个关于这些属性的数学函数式。

3.4 节展示了分类树与规则组的对应关系，其中，后者是数据的一种逻辑分类模型。而线性判别函数则是一种数值分类模型。比如，有一个特征向量 $\boldsymbol{x}$，其中每个特征元素为 x_i，则其线性模型可以写成公式 4-2 的形式。

公式 4-2：一个一般的线性模型

$$f(\boldsymbol{x}) = w_0 + w_1x_1 + w_2x_2 + \cdots$$

公式 4-1 若套入具体示例，则可以写成这种形式：

$$f(\boldsymbol{x}) = 60 - 1.0 \times \text{年龄} - 1.5 \times \text{账户余额}$$

要把这个模型作为线性判别式使用，在带入由特征向量 $\boldsymbol{x}$ 表示的实例时，我们需要判断 $f(\boldsymbol{x})$ 是正是负。上文中已说过，在二维空间中，这等同于判断实例 $\boldsymbol{x}$ 是在直线以上还是以下。

线性函数是数据科学的主力之一。我们终于开始涉及数据挖掘的话题了。现在已经有了**参数化**模型：线性函数的权重（w_i）即为参数。[1] 接下来，数据挖掘的任务就是用参数化模型来“拟合”某个特定数据集。具体来说，就是要找到这些特征的一系列最佳权重。

在完成模型学习后，这些权重往往也被宽泛地视作度量各特征重要性的指标。大体上，特征的权重越大，其对目标变量分类的重要性也越大（此处假设所有特征值都已经被归一化到了相同的取值范围，见本章补充栏“本章中的简化假设”）。同理，如果某个特征的权重趋近于 0，那么该特征往往可以被忽略或删除。目前我们关注的是，找到一组权重，而它不仅能足够正确地判别训练数据，还能尽可能精确地预测未知的目标变量。

然而，选择分类的最佳边界并非易事。如图 4-4 中的简单案例所示，图中的训练数据可以被一个线性判别式分类。但如图 4-5 所示，事实上有很多线性判别式可以完美地把类别划分开，它们的斜率和截距均不相同，而每一种组合代表一个不同的数据模型。实际上，能够对该训练集进行完美划分的线（或模型）有无穷种。那么，我们该如何选择呢？

注 1：为了防止该线穿过原点，一般需要引入权重 w_0，即截距。

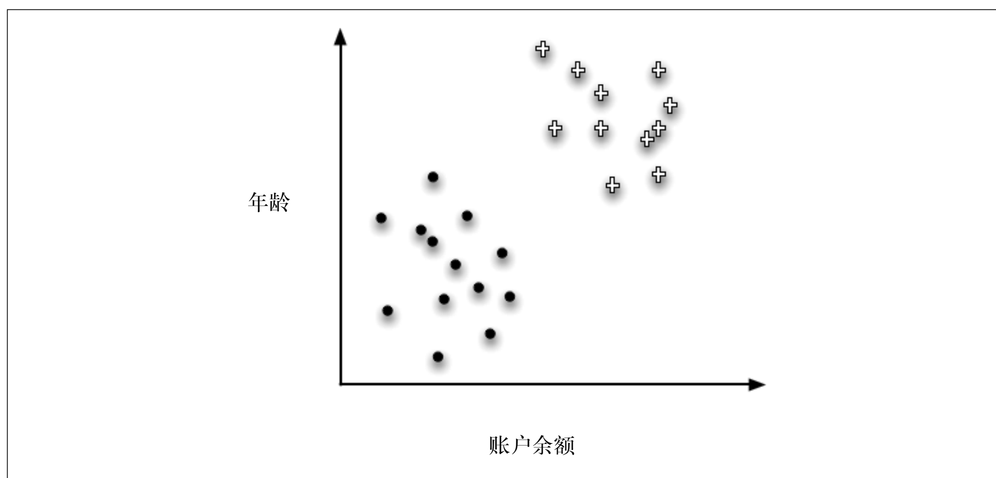


图 4-4：包含两个类的基础二维实例空间

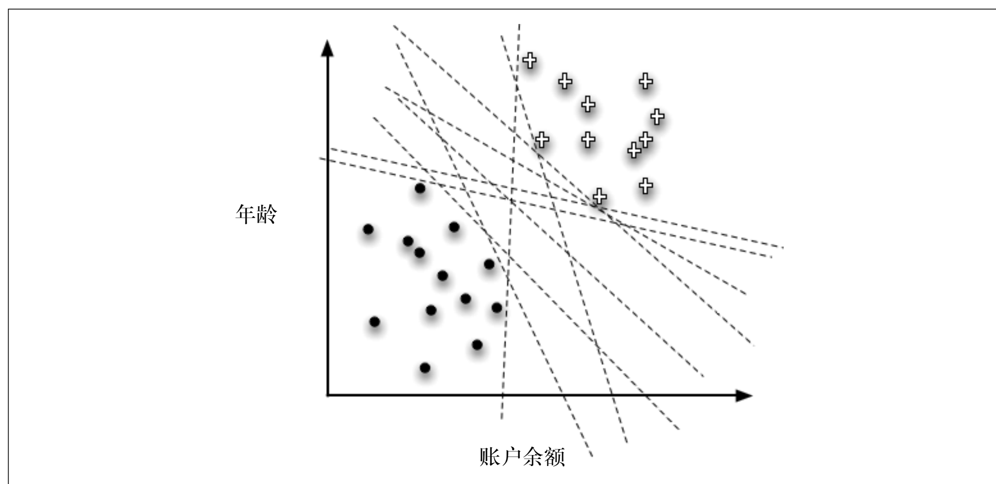


图 4-5：许多不同的线性边界都能对图 4-4 中的两类数据点进行恰当分类

4.1.2 目标函数的最优化

此处本章将引入数据挖掘中最重要的基本概念之一，也是一个连数据科学家们都经常忽视的概念：选择参数的目的，或者说**目标**，是什么？在该示例中该问题就转变为："我们需要选择哪种权重？"一般的做法是定义一个既能够体现目标，又能由一系列特定权重和一系列特定数据计算出的**目标函数**，然后通过最大化或最小化目标函数选出最优权重值。这里有一点很容易被忽略：只有当我们相信目标函数真正体现了目标时，或实际点来说，是所能找到的最佳替代品时，这些权重才真正是"最优的"。

不幸的是，要找到完全符合数据挖掘的真正目标的目标函数往往是不可能的。因此数据科

学家通常基于信念[2]和经验来选择目标函数。事实证明，有些选择非常有效。其中一种选择衍生出了所谓的“支持向量机”，本章将在展示一个简单目标函数的特定示例后简单提几句。然后，本章将稍微谈一下线性回归模型（而非分类模型），并以**逻辑回归**——最有用的数据挖掘技术之一——为结尾。“逻辑回归”这一名称有些用词不当，因为它执行的并非回归任务（即对数值型目标变量进行估计），而是把线性模型应用于类概率估计，而后者在许多情况下作用尤为突出。

线性回归、逻辑回归和支持向量机三者非常相似，都是用（线性）模型拟合数据这一基本技术的示例。它们的关键区别在于其目标函数各不相同。

4.1.3 示例：基于数据挖掘线性判别式

为了阐释线性判别函数，本章采用鸢尾花数据集的一个改编版本（http://archive.ics.uci.edu/ml/datasets/Iris）。该数据集取自 UCI 数据存储库（Bache & Lichman, 2013），是一个比较老但非常简单的数据集，描述了鸢尾花（一种开花植物）的不同种类。原始数据集包含三种鸢尾花的四种属性，而本章的数据挖掘任务就是根据这些属性来判断每个鸢尾花实例属于哪一种。

为便于讲解，本章只选取其中两种——**山鸢尾**和**变色鸢尾**。数据集包含了一组来自于这两种鸢尾花花朵的数据，每个花朵都由两种尺寸——“花瓣宽度”和“萼片宽度”（见图 4-6）——来描述。在该数据集的散点图（图 4-7）中，这两个变量分别由 x 轴和 y 轴代表。图中的每个散点代表了一朵花，即一个实例。其中“•”代表**山鸢尾**，“◦”代表**变色鸢尾**。

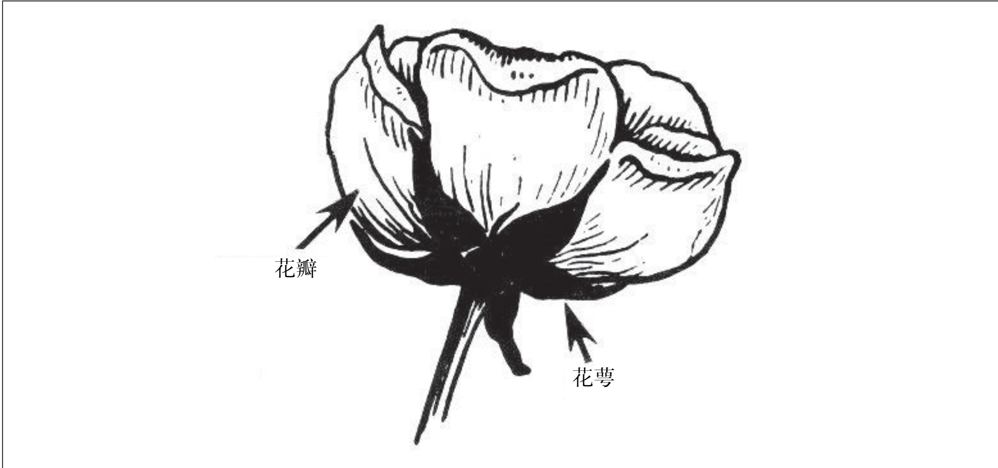


图 4-6：鸢尾花的两部分：花瓣和花萼。鸢尾花数据集包含花瓣和花萼宽度的测量值

注 2：有时候他们竟然很难承认这一点。

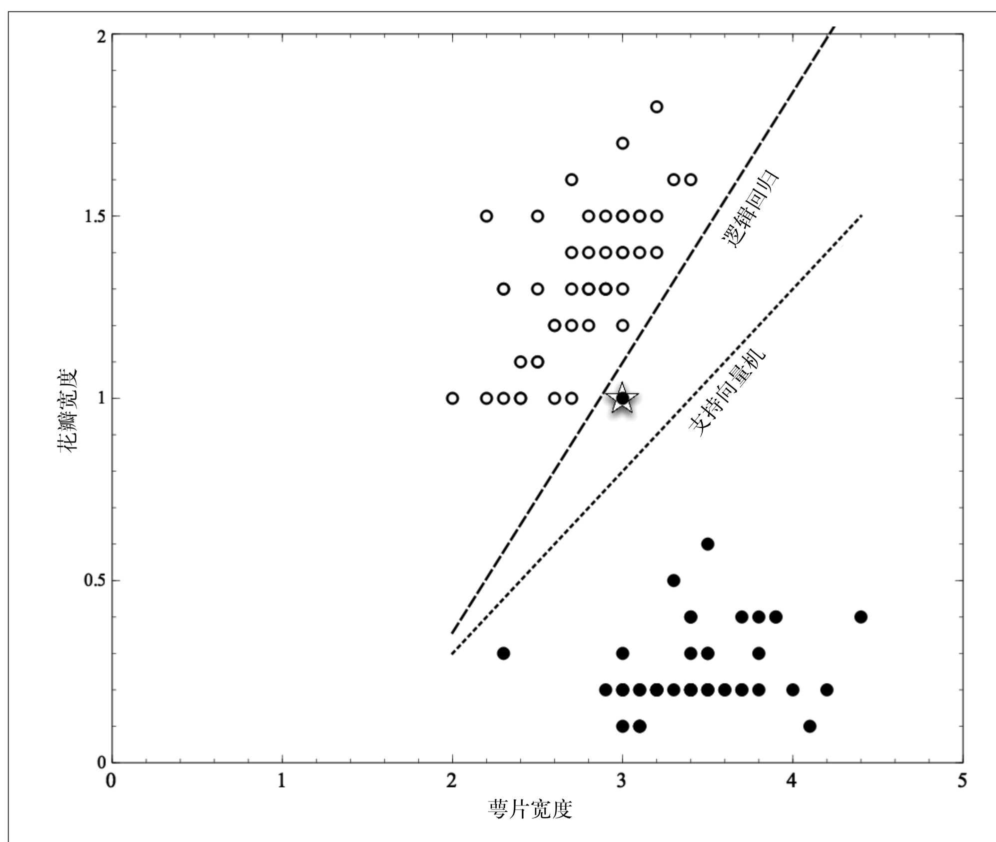


图 4-7：应用了两种线性分类器的数据集

图中呈现了两条不同的分割线，一条由逻辑回归生成，另一条由另一种线性方法——支持向量机（后文将简要探讨）生成。可以看到，该数据集中的数据点形成了非常集中的两簇和几个离群点。逻辑回归对这两个类别的数据点进行了彻底的划分，所有**变色鸢尾**实例都在线的左侧，而**山鸢尾**实例则都在右侧。支持向量机生成的分割线虽然几乎在两个簇的正中间，但是它将星标数据点 (3, 1) 分错了类[3]。你认为哪条分割线更好呢？我们将在第 5 章仔细学习它们产生差异的原因，以及两者的优劣。目前只需要知道，两者之所以产生了不同的边界，是因为它们对不同的目标函数进行了优化。

4.1.4　用线性判别函数对实例进行评分和排序

许多情况下，我们不仅想知道某个实例**是否**属于某个分类，还想知道哪些实例更有可能属于该分类。比如：哪些用户最可能对促销活动做出响应？哪些用户最有可能在合约到期后不再续约？解决以上问题的一个方法是，建立一个模型以输出类成员概率估计，就像第 3 章中用树型归纳进行类概率估计一样。另外，我们还可以用线性模型来处理该问题。本章

注 3：我们在原始数据集中加入了星标点，以强调两种分类方式产生的判别线的区别。

将在下文中引入逻辑回归时再详细探讨这一点。

在其他情况下，我们并不需要精确的概率估计，而仅需要评分。评分可以用来根据实例属于某一类的概率来对它们进行排序。比如，在目标市场营销中，因为用来对潜在客户进行营销的预算是有限的，所以我们想按照客户对营销活动做出响应的可能性对他们进行排序。在这种情况下，不需要取得精确的概率估计值，只需要这个排序足够合理，能使得排序中可能性最高的用户做出响应的可能性最大即可。

线性判别式可以轻易地给出这样的排序。观察图 4-4，假设“+”代表响应者，“•”代表不响应者。假设有一个新实例 $\mathbf{x}$，其类别未知（即我们还未对其提供特别活动），那么当其落在实例空间的哪一部分时，它最可能响应？落在哪一部分又表示它最不可能响应？而哪一部分是不确定区域？

许多人认为右侧贴近决策边界的部分是类别最不确定的区域（详见下文对“间距”的讨论），而远离决策边界的“+”区域是最可能响应的区域。根据上文给出的公式 4-2，当 $\mathbf{x}$ 落在决策边界上时（专业地说，即 $\mathbf{x}$ 是线上或超平面上一点时），$f(\mathbf{x})$ 为 0；当 $\mathbf{x}$ 接近边界时，$f(\mathbf{x})$ 相对较小；当 $\mathbf{x}$ 向“+”的方向远离边界时，$f(\mathbf{x})$ 为正且非常大。因此线性判别函数的结果 $f(\mathbf{x})$ 能根据属于某个类别的可能性，给出一个令人满意的直观排序。

4.1.5　支持向量机简介

虽然你如今仅仅接触了数据科学的边缘地带，但是终有一天你会碰到**支持向量机**（SVM）的概念。这个概念甚至会让许多数据科学界的大牛深感恐惧。这不仅仅是因为它的名字晦涩模糊，更是因为这种方法虽然原理让人难以理解，却非常有效。

幸好我们现在已经掌握了理解支持向量机所必需的概念。简而言之，支持向量机就是线性判别式。对许多与数据科学家打交道的商业用户来说，知道这一点就足够了。虽然如此，但我们可以更仔细地了解一下支持向量机。在了解了一些细节之后，我们就可以对拟合线性判别式的过程有一个直观上的认同。

和一般的线性判别式一样，支持向量机也依赖特征的线性方程（如公式 4-2）来对实例进行分类。



你可能也听过**非线性**的支持向量机。简而言之，由于非线性的支持向量机使用了不同的特征（原特征的函数），因而新特征的线性判别式就是原特征的非线性判别式。

因此，正如之前说过的，关键问题变成了：“用支持向量机拟合数据的目标函数是什么？”在这里，为了让读者获得直观的理解，本章先略过数学细节。这其中有两个主要概念。

回顾一下图 4-5，图中显示能对实例进行分类的线性判别式有无穷多种，而选择一个拟合数据用的目标函数，就相当于从图中选出分类效果最好的一条线。支持向量机的选择方法基于一个简单而巧妙的概念：先找出两类别间最宽的间距，而不是一条分类线，如图 4-8 中的平行虚线所示。

支持向量机的目标函数包含了“间距越宽越好”的概念。当最宽的间距被找到时，就把间

距的中心线作为线性判别式（见图 4-8 中的实心中线）。两条平行虚线之间的距离叫作线性判别式周围的**间距**，而我们的目标就是将该间距最大化。

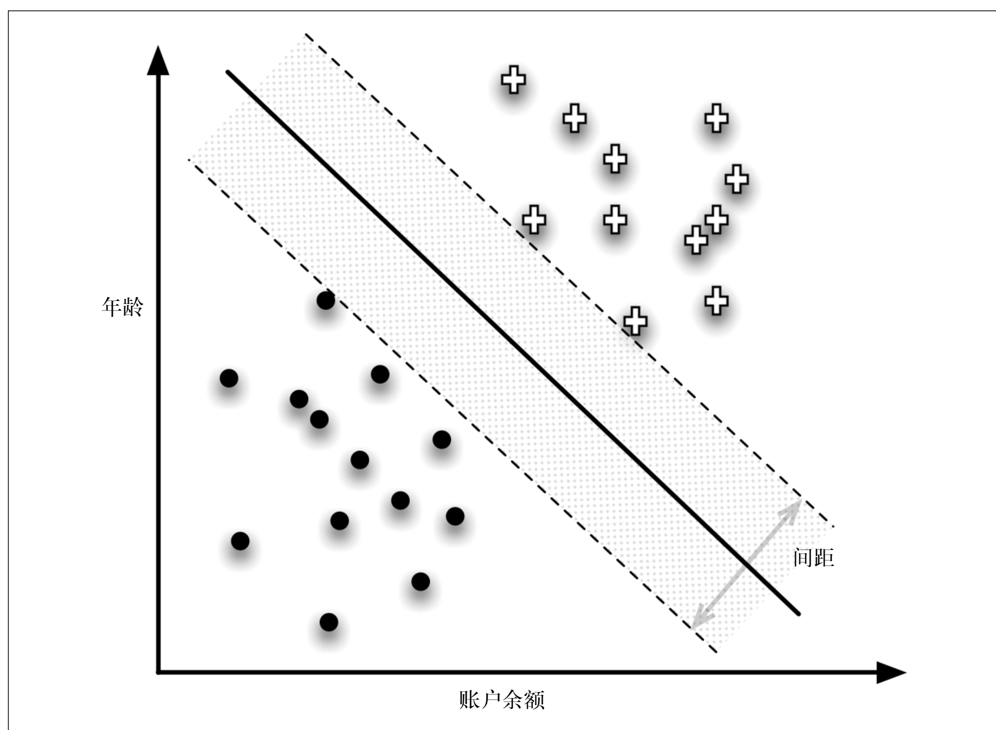


图 4-8：图 4-2 中的数据点和最大化的间距分类器

间距最大化这一概念之所以具有清晰直观的优势，其原因如下。训练集不过是来自于某个总体的样本，而在预测建模时，我们感兴趣的是预测未出现过的新实例的目标变量。我们会将这些新实例绘制成散点图，虽然这些新实例的分布很可能与训练集相似，但它们其实是不同的数据点。尤为突出的是，其中的一些正样本实例可能会比训练集中的任意一个正样本实例更靠近决策边界；同样，负样本也是如此。换句话说，这些实例可能会落在间距之中。间距最大化的分类决策边界恰恰为给这些点分类提供了最大的余地。具体来说，在使用 SVM 决策边界进行分类的情况下，某人如果想把新实例分入错误的类别，那么就必须将其置于间距深处任何其他线性判别式都无法到达的一点（或者干脆完全在间距的另一侧）。

支持向量机的第二条重要概念在于它对落在决策边界的错误一侧的数据点的处理方式。在图 4-2 的情况下，不存在能将所有数据点完美分类的单一直线决策边界。对大多数源于复杂的现实应用的数据来说，这很真实——许多数据点会不可避免地被模型分错类。但是，这并不意味着线性判别式不可靠，因为它不必把每个数据点都正确分类。然而，在用线性函数拟合数据时，我们不能仅仅从所有可以完美分类数据集的决策线中选一条，因为这样的完美分割线可能一条都不存在！

针对上述问题，支持向量机再一次给出了直观而令人满意的解决方案。跳过数学部分，其概念如下文所述。当使用目标函数来测量某个模型拟合训练集的效果时，我们会惩罚落入决策边界的错误一侧的数据点。如果数据线性可分，那么我们不会实施惩罚，而仅仅会使间距最大化；如果数据**并不**线性可分，那么其所能达到的最佳拟合即某个兼顾了较宽的间距和较低的总误差惩罚的平衡点。因为对分类错误的数据点的惩罚的大小与该点到间距边缘的距离成正比，所以支持向量机会尽可能只产生“小”误差。该误差函数被称作**合页损失**（见图 4-9 及 4.2 节中的“损失函数”）。

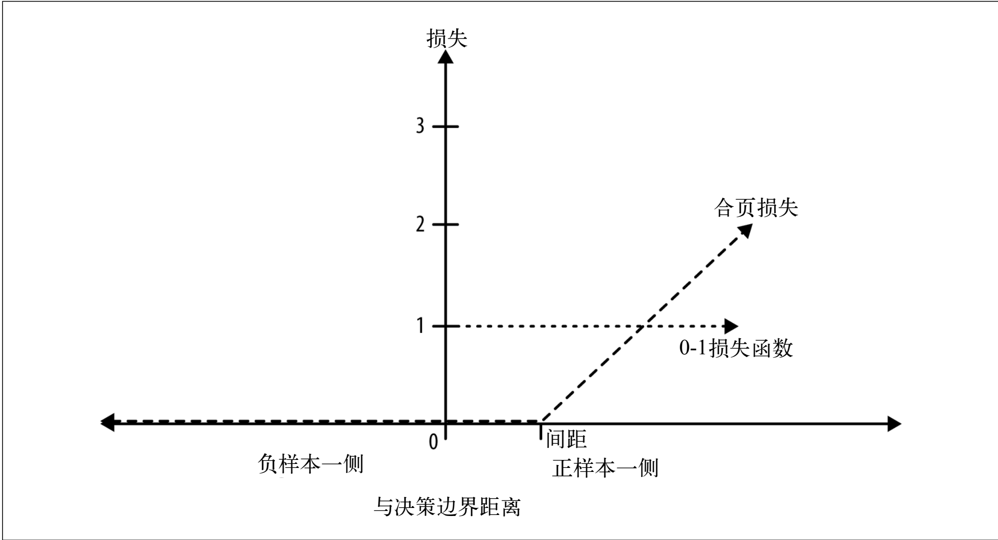


图 4-9：图中展示了两种损失函数。*x* 轴代表数据点到决策边界的距离，*y* 轴代表一个负实例引起的损失，它可以表示为关于该点到决策边界距离的函数（正样本实例的情况与之对称）。如果负样本实例落在边界的负样本一侧（预测正确），则不产生损失；如果它落在正样本一侧（即错误的一侧），则不同的损失函数会不同程度地惩罚它（见下一节中的“损失函数”）

4.2　通过数学函数进行回归

前一章介绍了选择富信息变量的基本概念，并且发现该概念同时适用于分类、回归和类概率估计。本章所讲的使用线性函数拟合数据的基本概念同样也适用于分类、回归和类概率估计。接下来，简单讨论一下回归。[4]

注 4：有关用于数据描述性分析的线性回归的文献浩如烟海，我们鼓励读者对此作深入研究。在本书中，我们仅把线性回归当作建模技术的一种。这样的方式的确可能与你学过的回归分析不同，因为我们关心的主要是线性回归的预测功能。其他作者则详细探讨了描述建模和预测建模的区别（Shemueli, 2010）。

损失函数

术语“损失”一般指误差惩罚，通用于数据科学领域。损失函数能够判断一个实例需要分摊多少惩罚。其判断基于模型预测值的误差——在当前语境中，即基于数据点到决策边界的距离。常用的损失函数有数种（图 4-9 体现了其中两种）。图 4-9 中，横轴代表数据点到决策边界的距离。分类错误的数据点到决策边界的距离为正，而分类正确的数据点到决策边界的距离则为负（图 4-9 中，对正负样本数据点的选择是随机的，对说明问题没有影响）。

支持向量机使用的是**合页损失**。我们之所以如此称呼它，是因为其损失图看上去很像合页。如果数据点没有落在间距的错误一侧，那么合页损失函数就不会给出惩罚。仅当数据点落在决策边界错误的一侧，且在间距边缘之外时，合页损失函数才为正。数据点到间距边缘的距离增加时，损失函数值随之线性增加。因此，数据点离决策边界越远，其受到的惩罚越多。

0-1 损失函数，正如其名，对正确决策的损失值赋值为 0，对错误决策的损失值赋值为 1。

为了进行对比，请想一想另一种形式的损失函数——**平方误差**。平方误差将数据点到决策边界的距离的平方定义为误差。它通常用于数值型预测（回归）而非分类，能极大地惩罚那些错得离谱的预测。而在分类问题中，它也能极大地惩罚落向“错误一侧”且远离决策边界的数据点。然而，平方误差同样会惩罚那些落向**正确**一侧且远离决策边界的数据点。因此在大多数商业问题中，选择平方误差作为分类问题或类概率估计问题的损失函数，有违“考虑损失函数是否与商业目标一致”的原则［针对这种不一致性，有人提出了合页版的平方误差（Rosset & Zhu，2007）］。

我们已经探讨了理解线性回归所需的绝大多数预备知识。线性回归模型的结构与公式 4-2 线性判别式函数式完全相同：

$$f(\boldsymbol{x}) = w_0 + w_1x_1 + w_2x_2 + \cdots$$

因此，根据参数化建模的总体思路框架，我们需要选出一个可以使模型拟合数据的效果最优的目标函数。可能的选择有多种。每个不同的线性回归建模过程都会使用一个不同的目标函数（而数据科学家必须仔细考虑它是否真的适合该问题）。

最常见的（即“标准的”）线性回归过程有非常强大且便捷的选择功能。回想一下，回归问题中的目标变量是数值型的。线性函数用公式 4-2 来给出目标变量的估计值，而**训练**数据中当然含有该目标变量的值。因此，关于模型拟合，我们凭直觉首先想到的是：估计值与训练集中的真实值的差异有多大？换句话说，模型拟合的误差有多大？假设要最小化该误差，针对一个给定的训练集，我们可以计算出每个数据点的误差并对这些结果求和。而其中误差和最小的模型就是拟合数据效果最好的模型。这也正是回归过程的做法。

你可能会注意到我们实际上还没确定目标函数，而这是因为计算估计值与真实值之间误差的方法有多种。其中最自然的方法就是用其中一个减去另一个，然后取结果的绝对值。因此，如果预测值为 10，而真实值为 12 或 8，那么误差就为 2。这被称作**绝对误差**，然后我

们可以将绝对误差的和最小化，或等价地将整个训练数据集的绝对误差的平均值最小化。这很容易理解，却不是标准线性回归过程的做法。

标准线性回归过程真正最小化的是这些误差的平方和或平均值，因此该过程也叫作“最小二乘”回归。为什么大众如此偏爱最小二乘法回归而很少考虑替代方法呢？原因只有两个字：方便。我们在基础统计学课程（及之后的课程）中就已学过该方法，而且对我们而言它触手可得，因为许多软件包中都装备了它。最初，最小二乘误差函数是由 18 世纪著名数学家高斯（Carl Friedrich Gauss）提出的，且其用法有理论依据的支撑（与正态分布，即高斯分布，有关）。更重要的是，从数学角度来说，平方误差非常方便。[5] 这在计算机出现之前对人们非常有帮助。从数据科学的角度来看，它用在理论分析方面也很方便，比如，它可以将模型误差根据原因清楚地分解开来。而分析师喜欢用平方误差的理由则更加现实：它能大大惩罚过大的误差。至于取误差的二次方作为惩罚是否合适，就要看特定的应用场景了。（为什么不对误差取四次方，以便更大程度地惩罚过大的误差呢？）

重要的是，任何目标函数都有自己的优势和劣势。最小二乘回归的一个严重缺陷是对数据过于敏感：误差点及其他离群点会大大扭曲最终得出的线性方程。在一些商业应用场景中，由于缺乏足够的资源，因而可能不能像在其他应用场景中一样，花费大量时间手动调试数据。在极端情况下，一些系统完全是自动构建模型并将它们投入应用的，因而其建模过程必须比详细的“手工”回归分析更加稳健才行。因此，在前一种应用场景中，我们需要更加稳健的建模方法（比如选择绝对误差而不是平方误差）。请谨记，一旦见到线性回归仅仅作为一个（线性）模型拟合数据的实例出现，我们就知道必须要选择合适的目标函数来进行最优化——而且做这些的时候，必须牢记最终的商业应用场景。

4.3 类概率估计和逻辑“回归”

之前提到过，在许多应用场景中，我们都需要估计新实例属于某个相关类别的概率。很多情况下，我们希望这些概率估计可以在与成本和收益等因素相关的辅助决策中发挥作用，比如，基于大量用户数据进行预测建模的方法已经广泛地被许多行业应用于欺诈检测，尤其是在银行业、电信业和电子商务业中。线性判别式可以用于判别某个账户或某笔交易中是否存在欺诈行为。而欺诈监控部门的主管可能不仅想知道哪些情况下出现欺诈的可能性最大，还想知道哪些情况下公司可能损失的钱最多（即哪些账户可能会给公司造成最大的金钱损失）。因此，我们需要估计欺诈的实际概率。（第 7 章将详细讨论商业问题中期望值的应用。）

所幸，在同样的线性模型拟合数据的框架下，通过选择一个不同的目标函数，我们可以设计出一个能给出精确类概率估计的模型。在完成上述任务的所有过程中，最常见的一种被称为逻辑回归。

注 5：有人对于这种选择的随机性表示反对，高斯对此表示认同。



类概率的精确估计到底是什么？这个话题超出了本书的讨论范围。大致上说，我们希望该概率估计是经过仔细校正的，比如，共有 100 个案例，其类概率估计为 0.2，那么其中大约有 20 个案例真的属于该类。我们还希望该概率估计有良好的区分能力，即能对不同实例给出有实际意义的不同的概率估计。后一个条件能避免模型只把“基础比率”（即总体的普遍率）作为对每个实例的预测。比如，总体中有 0.5% 的账户存在欺诈行为。如果不满足后一个条件的话，我们就可能会轻率地预测每个账户的欺诈概率均是 0.5%。这样的预测虽然是经过校正的，但完全没有区分能力。

为了理解逻辑回归，首先需要考虑：只用最基本的线性模型（公式 4-2）来预测类概率会出什么问题？前文已讨论过，直觉上来说和决策边界距离较远的数据点属于某一类（无论哪一类）的概率应该较高，而线性方程 $f(\mathbf{x})$ 的结果给出了这个距离值。然而，这同样暴露了问题所在：$f(\mathbf{x})$ 的值域是从 $-\infty$ 到 ∞，而概率的值域仅是 0 到 1。

所以我们要另辟蹊径。想一想：为了预测类成员的可能性，还有什么方法可以计算数据点到分割线的距离 $f(\mathbf{x})$。日常生活中是否存在其他表示可能性的概念？如果能想到取值为 $-\infty$ 到 ∞ 的概念，就能用线性公式为其建模了。

一个非常有用的替代概念是优势比，即某事件发生的概率与不发生的概率的比率。比如，如果某事件发生的概率是 80%，那么该事件的优势比就是 80：20 或 4：1。如果线性方程能给出优势比，那么只需稍稍进行代数运算就能得到事情发生的概率。接下来，请看一个更详细的示例。表 4-1 列出了不同的概率和其相应的优势比。

表4-1：概率和相应的优势比

概率	相应的优势比
0.5	50：50 或 1
0.9	90：10 或 9
0.999	999：1 或 999
0.01	1：99 或 0.0101
0.001	1：999 或 0.001 001

从表 4-1 中优势比的值域可以看出，该指标仍不能完全说明数据点到决策边界的距离。该距离的值域是从 $-\infty$ 到 ∞，而该例中的优势比值域则是从 0 到 ∞。尽管如此，通过对优势比取对数（称作“对数优势比”），我们可以轻松地解决这个思维惯性问题，因为任何非负数取对数之后都有可能变为负数，见表 4-2。

表4-2：概率、优势比和对应的对数优势比

概率	优势比	对数优势比
0.5	50：50 或 1	0
0.9	90：10 或 9	2.19
0.999	999：1 或 999	6.9
0.01	1：99 或 0.0101	–4.6
0.001	1：999 或 0.001 001	–6.9

因此，如果不想对类成员概率估计建模，而仅想对可能性的某个概念建模，那么我们可以使用 $f(\boldsymbol{x})$ **对数优势比**进行建模。

你看，我们另辟蹊径最终却回到了本章的主题。这就是一个逻辑回归模型：本章通篇考察同一个线性方程 $f(\boldsymbol{x})$，而它被用来测量相关“事件”的对数优势比。更具体地说，$f(\boldsymbol{x})$ 是模型所估计 $\boldsymbol{x}$ 属于正向一类的对数优势比。比如，模型可以估计在合约到期时，一个由特征向量 $\boldsymbol{x}$ 描述的用户会离开公司的对数优势比。经过简单代数运算，我们就能把该对数优势比转换为类别成员概率。这一点比本书大部分内容技术性略强些，因此本章把它归入了特殊的“技术细节”小节（下文），该小节还讨论了用逻辑回归拟合数据时所需要的最优化的目标函数。你可以详细阅读该小节，也可以略过，其要点如下。

- 针对概率估计，逻辑回归使用了线性模型，而该模型同样可以用于线性判别式的分类问题和预测数值型目标变量值的线性回归问题。
- 逻辑回归模型的输出结果可以理解为类成员的对数优势比。
- 由于这些对数优势比可以直接转化为类成员概率，因而逻辑回归往往只被当作一种类别概率的模型。你肯定在不自知的情况下接触了许多逻辑回归模型，因为它们被广泛地应用于数量估计中，如信贷违约概率、对活动做出响应的概率、账户欺诈概率、文章的主题归属概率等。

在探讨过技术细节之后，我们将会对本章的线性模型和第 3 章的树形结构模型进行对比。



“逻辑回归”是误称

上文提到，在数据科学术语的现代用法中，“逻辑**回归**”其实用词有误。分类问题和回归问题的区别在于目标变量是类别型还是数值型。逻辑回归进行的是数值预测（即对对数优势比进行预测）。但是，数据中的目标变量却是类别型。有关这一点的讨论学术性非常强。重要的是理解逻辑回归的功能。它所估计的是对数优势比，或更宽泛地，一个类别型变量中的某个类别值的概率（数值）。因而尽管它名称如此，我们还是认为它是一个类概率估计模型，而**非**回归模型。

*逻辑回归：一些技术细节



前方有技术细节！

由于逻辑回归应用如此广泛，且不像线性回归一样直观，因而我们需要介绍一些技术细节。但跳过此节也并不影响你对其他章节的理解。

那么严格地讲，逻辑回归模型的底线到底是什么？

设 $p_+(\boldsymbol{x})$ 为模型对由特征向量 $\boldsymbol{x}$ 描述的数据点的类别概率的估计[6]，设类别“+”为任意一个我们想对其建模的（二元）事件：如对优惠活动做出响应、合约到期后不再续约、遭受欺

注 6：通常我们在技术处理中用“^”的记号（如 $\hat{p}$），将模型的类别概率估计值和类别的真实概率区分开来。虽然本书中不会使用该记号，但技术控读者们需要将该记号谨记于心。

诈等，则该事件不发生的概率就是 $1-p_+(\boldsymbol{x})$。

公式 4-3：对数优势比线性函数

$$\log\left(\frac{p_+(\boldsymbol{x})}{1-p_+(\boldsymbol{x})}\right)=f(\boldsymbol{x})=w_0+w_1x_1+w_2x_2+\cdots$$

公式 4-3 表明，对于一个由特征向量 $\boldsymbol{x}$ 描述的数据项而言，其类别的对数优势比等于线性函数 $f(\boldsymbol{x})$ 的值。由于我们想要的通常是类别概率的估计值（而非对数优势比），因而可以由公式 4-3 解出 $p_+(\boldsymbol{x})$，从而得到公式 4-4 中这个不大好看的量。

公式 4-4：逻辑函数

$$p_+(\boldsymbol{x})=\frac{1}{1+\mathrm{e}^{-f(\boldsymbol{x})}}$$

尽管公式 4-4 中的量不大好看，然而以特定的某种方法绘出这个公式之后，就可以发现，它与我们直觉上的认识非常相符：远离决策边界的类成员估计较为确定，而靠近决策边界的则较为不确定。

在图 4-10 中，概率估计值（$p_+(\boldsymbol{x})$，纵轴）是数据点到决策边界的距离（横轴）的函数。如图所示，在决策边界上（$x = 0$），概率为 0.5（相当于丢硬币）；在决策边界附近，概率几乎呈线性变化；而距离决策边界越远，概率的确定性越高。模型“拟合”数据的过程包括了确定这段近似线性的部分的斜率，由此我们确定了在远离决策边界时，分类的确定性增加得有多快。

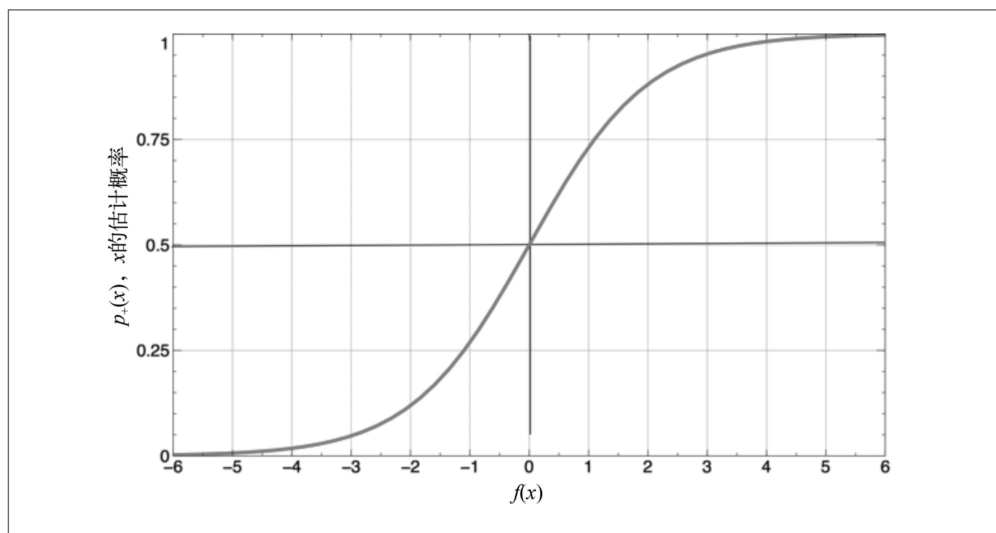


图 4-10：逻辑回归以 $f(x)$（数据点到决策边界的距离）的函数进行类概率估计。该曲线之所以被称作“S 形函数”，是因为它是 S 形的。它能将概率置于正确的取值范围（0 到 1 之间）

正文跳过的另一个主要技术点是：在用逻辑回归模型拟合数据时使用的目标函数是什么？

之前提过，训练集的目标变量是二元的。我们可以将逻辑回归模型应用于训练数据，并估计出训练数据中属于目标类的每个数据点。我们想要的理想情况是，对所有正实例 $\boldsymbol{x}_+$，都有 $p_+(\boldsymbol{x}_+) = 1$；而对所有负实例 $\boldsymbol{x}_\bullet$，都有 $p_+(\boldsymbol{x}_\bullet) = 0$。可惜在实际情况中，我们很难完美地估计这些概率（想一想根据用户资料估计哪些用户会对某个促销活动做出回应的问题，你就明白了）。尽管如此，我们还是希望 $p_+(\boldsymbol{x}_+)$ 尽可能接近 1，而 $p_+(\boldsymbol{x}_\bullet)$ 尽可能接近 0。

这引出了逻辑回归模型拟合数据时所使用的标准目标函数。若有一系列可以产生类概率估计 $p_+(\boldsymbol{x})$ 的参数 w，那么就有下面这个函数。它可以用于计算某有标注的实例属于正确分类的“可能性”，请思考一下它：

$$g(\boldsymbol{x},w)=\begin{cases}p_+(\boldsymbol{x}) & \text{若}\ \boldsymbol{x}\ \text{为}+\\ 1-p_+(\boldsymbol{x}) & \text{若}\ \boldsymbol{x}\ \text{为}\bullet\end{cases}$$

函数 g 能根据 $\boldsymbol{x}$ 的特征估计 $\boldsymbol{x}$ 的实际类别概率。现在我们对有标注的数据集中的所有数据点的 g 值求和，然后对不同的参数化模型（本例中即逻辑回归的不同权重集合）重复这个计算。因为给出最大 g 值汇总的模型（权重集合），其给出的数据的“似然性”也最大，所以其又称“最大似然模型”。最大似然模型“通常”会对正样本实例给出最高的概率，而对负样本实例给出最低的概率。

类标签和概率

你可能认为目标变量就是类成员概率，而训练数据中目标变量的观测值仅仅会在实例的观测值为该类时令 $p(\boldsymbol{x}) = 1$，不为该类时则令 $p(\boldsymbol{x}) = 0$。然而，这和逻辑回归模型的用法不同。以目标市场营销中的某个应用场景为例。对用户 c 而言，模型可能会预测他响应某促销活动的概率是 $p(c\ \text{响应}) = 0.02$。但在数据中，我们发现该用户确实响应了该促销活动。这既不意味着该用户响应的概率实际上是 1.0，也不意味着模型犯了致命错误。这是因为，用户的响应概率可能的确在 0.02 左右，而这实际上对于许多活动而言已经很高了，只是用户这一次碰巧的确做出了响应。

另一种更好的思路是，虽然训练数据集包含对潜在概率的一组统计“提取”，但它不是潜在概率本身。而后逻辑回归过程会使用线性 – 对数优势比模型对这些概率（实例空间中的概率分布）进行估计。该估计即基于上述分布的提取结果的观测数据。

4.4　示例：对比逻辑回归和树型归纳

尽管分类树和线性分类器都使用了线性决策边界，然而两者仍有两个重要区别。

(1) 分类树使用的决策边界与实例空间的坐标轴垂直（见图 4-1），而线性分类器所使用的决策边界的方向是任意的（见图 4-3）。这是因为分类树每次只选择一个属性，而线性分类器使用的则是所有属性的加权组合。

(2) 分类树是个“分段式”分类器，在必要时会用分而治之的方法对实例空间进行递归式划分。原则上，分类树可以随意将实例空间反复切分，直到它变成极小的区域（尽管第 5

章会谈到为何要避免这么做)。而由于线性分类器仅把一个决策平面放置在实例空间中，故而它可以自由选择方向。但该优势仅限于把实例空间一分为二的情况。这是因为决策平面是一个包含了所有变量的（线性）公式，而该公式必须适合整个数据空间。

对于给定数据集而言，事先确定其最佳的变量组合往往并不容易。你很可能不会知道最佳决策边界是什么样的。那么这些区别实际上产生了什么结果呢？

在将模型应用到商业问题中时，不同背景的企业利益相关者会对模型产生不同的理解，比如，对统计学知识背景较强的人而言，逻辑回归的作用非常易于理解；但对没有这类知识储备的人而言，它却非常晦涩难懂。但是只要不过于庞大，对于统计学或数学知识背景较弱的人而言，决策树理解起来要容易得多。

为什么理解这些区别如此重要？在许多商业问题中，数据科学团队无法最终决定使用或者部署哪个模型。通常会有至少一个管理者来“批准”模型的使用，而且在许多情况下需要有好几个利益相关者认可该模型。比如，如果要通信公司部署一个新模型，以便于派遣技术人员为呼叫客服的用户做维修，那么运营支撑部门、客户服务部门和技术开发部门的管理人员都需要认同新模型利大于弊——因为对于该问题而言，完美的模型是不存在的。

我们先在一个简单的真实数据集中试用一下逻辑回归（http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)）。该数据集来自威斯康星州乳腺癌数据集。和几节之前的鸢尾花数据集和上一章的蘑菇数据集一样，这个数据集也来自加州大学欧文分校的机器学习数据仓库。

其中，每个实例都描述了一幅细胞核图像的特征。而且根据专家诊断，它们被标记为**良性细胞**或**恶性细胞**（癌细胞）。图 4-11 展示了一个细胞图像样本。

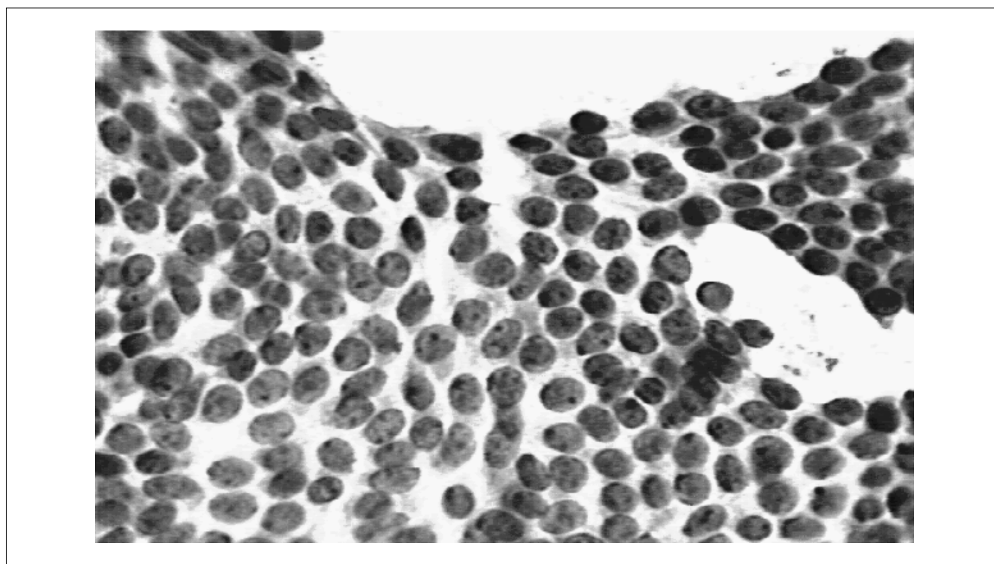


图 4-11：威斯康星州乳腺癌数据集中的一幅细胞图（图片由 Nick Street 和 Bill Wolberg 提供）

每张图中，有 10 个基本特征被提取出来，如表 4-3 中所列。

表4-3：威斯康星州乳腺癌数据集中的属性

属性名	描　　述
半径	中点到周长的平均距离
纹理	灰度值的标准差
周长	细胞集团的周长
面积	细胞集团的面积
平滑度	半径长度的局部变化
紧密度	计算公式为：半径 ^{2}/ 面积 −1.0
凹度	轮廓的凹陷程度
凹点	轮廓凹陷部分的数量
对称性	细胞核对称性指标
分形维数	“海岸线近似”−1.0
诊断（目标变量）	细胞样本的诊断结果：恶性 / 良性

以上变量“由乳腺肿块的细针抽吸（FNA）数字化图像计算得出，描述了图中细胞核的特征”。以下计算了其中每个基本特征的三个值：均值（_mean）、标准差（_SE）和“最差值”或最大值（三个最大值的均值，_worst），得到了 30 个测量属性。共包含 357 幅良性细胞图像和 212 幅恶性细胞图像。

表 4-4 展示了基于该数据集通过逻辑回归学习得到的线性模型。它可以用于预测癌细胞是良性还是恶性。比较突出的一点是，它会把非 0 的权重按从高到低进行排序。

表4-4：对威斯康星州乳腺癌数据集进行逻辑回归得到的线性公式（变量的描述可见正文及表4-3）

属　　性	权重（学习后得到的参数）
平滑度 _ 最差值	22.3
凹点 _ 均值	19.47
凹点 _ 最差值	11.68
对称性 _ 最差值	4.99
凹度 _ 最差值	2.86
凹度 _ 均值	2.34
半径 _ 最差值	0.25
纹理 _ 最差值	0.13
面积 _ 标准差	0.06
纹理 _ 均值	0.03
纹理 _ 标准差	–0.29
紧密度 _ 均值	–7.1
紧密度 _ 标准差	–27.87
w_0（截距）	–17.7

该模型的效果还不错，在整个数据集中只预测错了 6 个数据点，准确率约为 98.9%（预测正确的数据点所占的比例）。为了做比较，本章根据同一个数据集学习得到了其分类树（使用了 Weka 的 J48 算法），该分类树参见图 4-12。这棵树共有 25 个节点，其中叶节点有 13 个，这就意味着分类树把所有实例划分为了 13 个分组。该分类树的准确率为 99.1%，略高于逻辑回归。

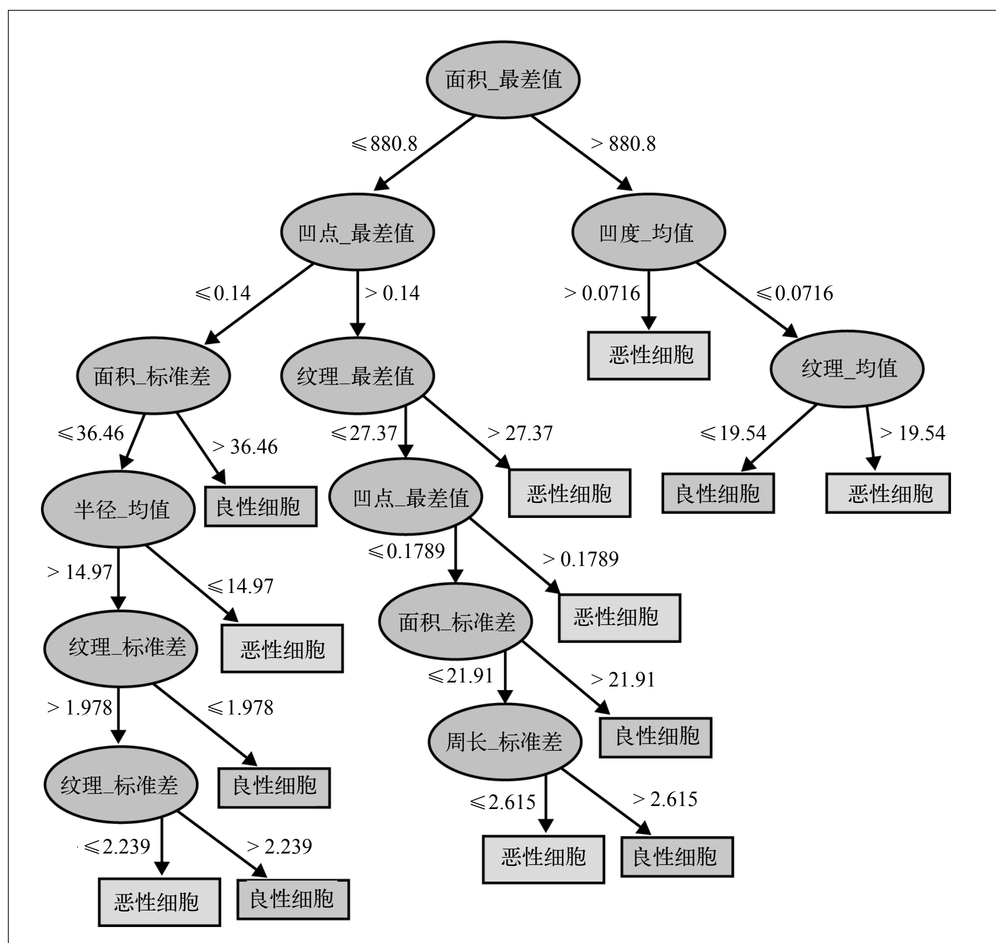


图 4-12：对威斯康星州乳腺癌数据集进行学习得到的决策树

虽然这次实验仅是为了展示对同一个数据集采用两种方法的不同结果，但我们也有必要暂时偏离主题，对这些结果稍加思考。首先，98.9% 的准确率听起来不错，可现实中真能如此吗？虽然在数据挖掘文献中，这样的数字屡见不鲜，但在现实问题（如癌症诊断）中，对分类器的评估往往非常困难、非常复杂。本书将在第 7 章和第 8 章中详细探讨该评估问题。

其次，想一想这两种方法的结果。它们的准确率分别为 98.9% 和 99.1%。因为分类树的准确率略胜一筹，所以我们很可能会认为这个模型更好。但这种想法正确吗？这点细小的差异仅仅是由 569 个数据点中的一个产生的误差引起的。况且，这些准确率是通过评估其各自的模型得出的，而评估模型和构建模型使用了相同的实例集。这种评估的可信度又是多少？第 5 章、第 7 章和第 8 章将对模型评估的准则和缺陷进行探讨。

4.5 非线性函数、支持向量机和神经网络

目前为止，本章集中讨论了数据科学中最常用的数值函数：线性模型。线性模型包含了种类繁多的技术。另外，如图 4-13 所示，**如果我们把更复杂的特征纳入线性函数中**，那么就可以用这样的函数来体现非线性模型。本例使用了 4.1.3 节中的鸢尾花数据集，并在输入数据中加入了**萼片宽度的平方**这一平方项。这样得到的模型是原特征空间中的一条曲线（抛物线）。本例还在原数据集中加入了一个数据点，即坐标为 (4, 0.7) 的变色鸢尾实例，并用星号标注。

本书的基本概念比应用线性函数拟合要广泛得多。当然，我们可以设定任意的复杂数值函数，并用其参数拟合数据。基于拟合复杂非线性函数的各类技术中，最常用的两类被称作**非线性支持向量机**和**神经网络**。

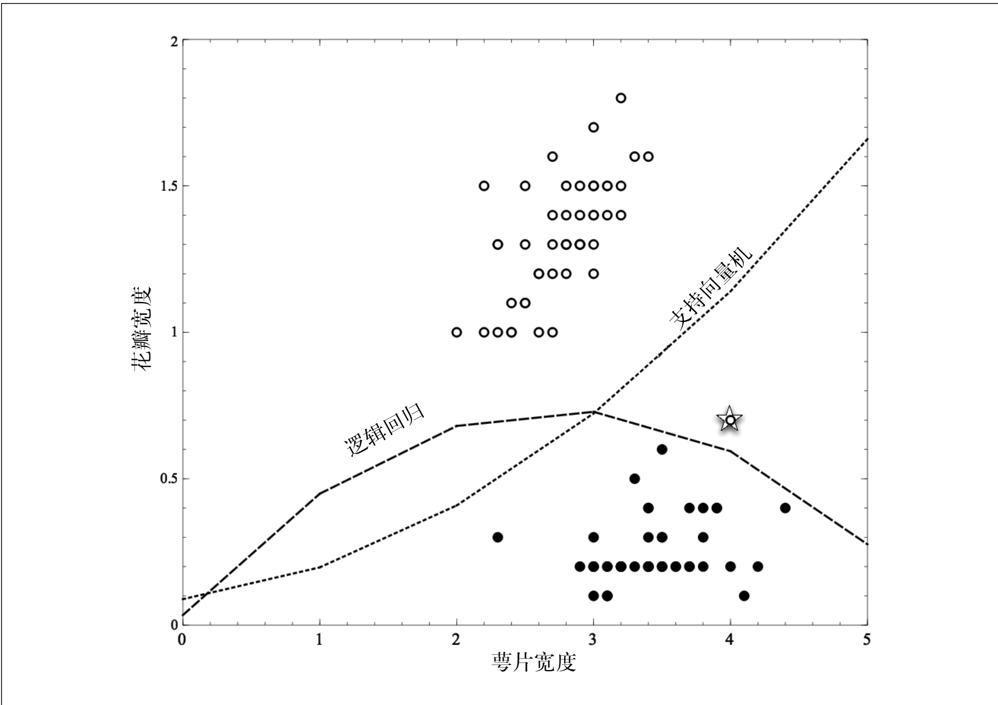


图 4-13：包含非线性特征的鸢尾花数据集。图中，我们在逻辑回归和支持向量机（两种线性模型）中加入了一个特征——萼片宽度的平方。这样，如图所示，两种模型都转变成了了复杂的非线性模型（即非线性决策边界）

本章讨论了如何在模型中加入复杂变量并用线性函数来拟合它的“技巧”，而本质上，你可以把非线性支持向量机看作系统性地实现这个“技巧”的方法。支持向量机有一个所谓的“核函数”，它能将原始特征映射到其他特征空间中，随后我们可以用线性模型拟合这个新的特征空间，如图 4-13 中示例所示。推而广之，我们可以使用一个包含“多项式核函数”的非线性支持向量机，本质上也就是说，我们可以使用一个原初特征的“高阶”组合（譬如，特征的平方项、特征的乘积）。数据科学家应该熟悉各种形式的核函数（线性、多项式型等）。

根据本章的基本概念，神经网络同样能执行复杂的非线性数值函数。神经网络有一个有趣的不同之处。你可以把神经网络想成“一叠”模型，而它的最底层是原始特征。通过这些特征可以得到许多相对简单的模型，姑且先假设它们为逻辑回归模型。而其上每一层都会输出一个简单的模型（仍然假设为逻辑回归模型），并将其用于输出再上一层的模型。因此，在一个两层的模型组合中，我们可以根据原始特征学习得到一个逻辑回归模型，再把该逻辑回归模型的结果作为下一个待学习的逻辑回归模型的特征。我们可以把这个过程粗略地想成，首先针对问题的不同角度创建一系列的“专家”（第一层模型），然后考虑如何把这些专家的意见按不同权重进行组合（第二层模型）。[7]

神经网络的概念愈发有趣了。我们可能会问：如果需要用低层的逻辑回归——即不同的“专家”——进行学习，那么每一个逻辑回归的**目标变量**是什么？在构建这种层叠模型时，有些人用特定的目标变量来构建代表特定事物的低层“专家”（如 Perlich 等，2013），但更一般性的方法是，神经网络的训练目标标签仅对最终一层使用（即实际的目标变量）。那么，如何训练低层的逻辑回归呢？此时需要回到本章的基本概念上。一叠模型可以表达为一个大型的参数化数值函数，其参数就是所有模型的系数。因此，一旦决定了用哪个目标函数来表达所希望优化的内容（比如基于某些拟合函数拟合训练数据的效果），我们就可以应用最优化过程，为这个非常复杂的数值函数找出最佳参数。在完成这个过程之后，我们会同时得知所有模型的参数、低层“专家”的最佳参数，以及组合这些模型的方法。



神经网络适用于多种任务

本节描述了用于分类任务和回归任务的神经网络。神经网络这一领域博大精深，历史悠久，且在数据挖掘中应用广泛。第 2 章提及的许多任务都使用了神经网络，如聚类、时间序列分析、画像分析等。

既然神经网络听起来这么酷，我们为什么不一直用它呢？需要权衡的是，我们在提高模型拟合的灵活性时，也会提高对模型拟合得**过好**的可能性。可能出现的情况是，模型能拟合特定的训练集中的细节，却不能找出适用范围更广的模式或模型。尤其是，我们希望模型不仅适用于目前的训练集，还能适用于来自同一个总体或者应用场景的其他数据集。这种考虑不是仅仅针对神经网络，而是广义地针对所有模型。这是数据科学领域中最重要的概念之一，也是下一章的主题。

注 7：可以将其与第 12 章的**集成方法**的概念相比较。

4.6 小结

本章介绍了第二种预测建模技术，它被称为“函数拟合”或“参数化建模”。在这种情况下，模型是一个部分确定的公式——一个由数据中属性定义的、某些数值参数未定的数值型函数。数据挖掘过程的任务就是通过找到一个（某种意义上的）最佳参数组合以使模型“拟合”数据。

虽然函数拟合技术多种多样，但是它们大部分使用同一种线性结构的模型：属性值的简单加权求和。而属性的权重就是数据挖掘所要拟合的参数。线性模型技术包括了传统线性回归、逻辑回归和诸如支持向量机的线性判别式。从概念上来说，这些技术的关键区别在于其对一关键问题——**对数据的最佳拟合究竟是指什么**——的不同回答。拟合效果好坏往往由“目标函数”描述，不同的技术使用的目标函数不同，因此作为结果的技术之间也存在巨大差异。

我们已经学习了两种截然不同的建模方法：树型归纳和函数拟合，并对两者进行了比较（见 4.4 节）。本书还引入了两种评估模型的标准：模型的预测效果和模型的可理解性。为了更加了解数据集，更好的方法是尝试在同一个数据集内用多种方法建模。

本章集中讨论了“模型拟合数据效果最优化”这一基本概念。然而，这也会引出数据挖掘中最重要的基础**问题**——如果花足够的功夫，那么你总能在数据集中找到结构，哪怕这样的结构只是偶然出现。这样的趋势被称作**过拟合**。识别和避免过拟合是数据科学中一个重要的主题，将在下一章探讨。

第5章

避免过拟合

基本概念：泛化能力；拟合和过拟合；复杂度控制

示例方法：交叉验证；属性选择；剪枝；正则化

数据科学中最重要的基本概念之二就是过拟合和泛化能力。如果在某个数据集中寻找模式时足够灵活，那么我们总能找到一些模式。然而，这些所谓的“模式”可能仅仅是偶然出现在数据中。正如前文所提到的，我们想要得到泛化能力更强的模式，即能很好地预测尚未观测到的实例的模式。若在数据集中发现了看起来非常好的“模式”，但是这个“模式”事实上只是偶然出现，不具有普遍的适用性，那么这种情况就称为对数据的**过拟合**。

5.1　泛化能力

请考虑下面这个（极端的）例子。假如你是 MegaTelCo 的经理，负责降低用户流失率，而我是某数据挖掘咨询团队的主管。你给我的团队提供了一个历史数据集，其中包括合约到期后六个月内仍留存的用户的历史数据和流失的用户的历史数据。我的工作就是像前文提到的那样，构建一个基于用户特征来判断哪些用户可能会流失的模型。我通过挖掘数据构建了一个模型，并把模型的代码交给你，以便你将该模型部署到公司用于降低用户流失率的系统中。

当然，你非常想知道这个模型的效果如何，于是让技术团队用历史数据对模型进行检验。你知道历史表现良好不等于未来也能取得成功。但经验告诉你，除非行业中出现大的变动（如 iPhone 的推出），否则用户流失的模式一般是稳定的。而且你知道从数据收集完到现在，并没有发生过这样的变动。于是，技术团队用历史数据检验了模型。技术主管报告说该模型的效果好得惊人：模型准确率为 100%，它对所有流失用户和未流失用户都进行了正确分类，没有做出一次误判。

但是，经验丰富的你对这个结果并不满意。你已经让专家们观察用户流失行为很久了，如果真的存在百分之百精确的预测指标，那么你做得应该会比现在好得多。也许这只是运气好？

其实并非如此。我们的数据科学团队可以让每次测试都达到这种效果，其建模过程如下。先把每个流失用户的特征向量存储在一个数据库表中，将其命名为 T_c。在使用过程中，当模型被用于判别某用户流失的概率时，它会提取该用户的特征向量，并在 T_c 中对其进行查找，如果找到，则显示“流失概率为 100%”；如果没找到，则显示“流失概率为 0%”。因此，当技术团队把该模型应用到历史数据中时，就会发现该模型完美预测了所有情况。[1]

这个简单方法叫**表模型**。它能记住训练集却毫无泛化能力。这会导致什么问题呢？想想在实践中该怎么使用这个模型。当一个**之前没出现过的**用户的合约快到期时，我们想使用这个模型预测其流失的概率。因为历史数据集中不包含该用户，所以模型找不到这个用户的特征向量，于是就会报告“流失概率为 0%”。实际上，该模型会对所有（不在训练集中的）用户做出这样的预测。这个模型看似完美，但在实践中却毫无用处！

这个场景看起来可能很荒谬，因为在现实中没人会把原始的用户数据往表里一存，就声称其是某事件的“预测模型”。但我们需要思考这种做法为何不正确，因为它预测失败的原因和其他现实中的数据挖掘工作失败的原因是相同的。这个极端例子包含两种相关的数据科学基本概念：**泛化能力**和**过拟合**。泛化能力是模型本身或建模过程的一种性质，具备这种性质意味着模型可以被应用到建模数据集以外的数据上。而这个示例中的模型却无法应用到其他数据上。它是为原始数据集量身定做（或“完全拟合”）的。这种情况其实就是“过拟合”。

明白这一点其实非常重要，因为所有数据集都是总体的一部分。在这个示例中，样本来自于手机用户这一总体，而我们希望模型不仅能应用于训练集，还能推广到整个总体。有时我们会担心训练集不能很好地代表整个总体，但本例的问题却不在于此。其问题在于，虽然训练数据具有代表性，但是数据挖掘却没能从中构建出一个具有泛化能力的模型。

5.2 过拟合

过拟合是一种数据挖掘过程牺牲模型对新数据点的泛化能力，从而使其完美适用于训练集数据的倾向。前文中的示例其实有些勉强，因为其中的建模过程完全依赖于记忆功能，而这是过拟合的最极端情况。然而，所有数据挖掘过程或多或少都可能出现过拟合的情况。如果我们仔细观察数据，那么总能从中找到各种模式。正如诺贝尔奖获得者 Ronald Coase 所说：“如果你拷问数据的时间足够长，那么它总会招供的。”

糟糕的是，这个问题是潜在的。而其解决之道既不是强求一个绝对不存在过拟合的模型，因为所有模型都存在这个问题；也不是单纯追求过拟合程度较轻的模型，因为我们需要权衡模型的复杂度和过拟合的概率。有时我们可能想要更复杂的模型，因为它们可以更好地刻画应用场景中的实际复杂度且更加精确。没有任何一个选项或过程能够消除过拟合，最

注 1：严格来说，这不能百分之百实现，因为数据中可能存在两个特征向量相同的用户，其中一个流失，而另一个却未流失。但在本例中可以忽略这种可能性。比如我们可以假设唯一的用户 ID 也是特征之一。

好的方法就是有原则地识别过拟合和控制复杂度。

本章后面会进一步探讨过拟合、评估模型过拟合程度的方法，以及如何尽可能避免过拟合。

5.3 过拟合检验

在讨论如何处理过拟合之前，我们需要先知道如何识别过拟合。

5.3.1 保留数据和拟合图

本节将介绍一个简单的分析工具——**拟合图**。拟合图能以复杂度函数的形式展示模型的准确率。为了检验过拟合，还需要引入数据科学中对评估很重要的一个概念——**保留数据**。

前文中示例的问题在于，评估模型时用的是训练集，也就是用来建模的数据，而这无法评估模型对未出现过的数据的泛化能力。因此，我们需要"保留"一些目标变量值已知，却没有用来建模的数据。这些数据并非最终用来预测目标变量值的数据，而是用来评估模型泛化能力的数据。这种做法相当于实验室测试。我们会用保留数据模拟使用场景，对模型（甚至是建模人员）隐藏保留数据的目标变量值，然后用模型进行预测。之后，再通过比较模型预测值和真实值来评估模型的**泛化能力**。模型在训练集上的准确率（有时被称作"样本内"准确率）和在保留数据集上的准确率之间很可能存在差异，因此此处的保留数据通常被称作"测试集"。

模型的准确率取决于其复杂度，而复杂度体现在多个方面，本章稍后会对其进行探讨。我们先用训练集和保留数据集的区别来更准确地定义拟合图。拟合图（见图 5-1）展现了随着模型复杂度的改变，其应用于训练集和保留数据集时准确率的差异。一般情况下，模型越复杂，过拟合的情况就越严重。（从技术上讲，建模过程越灵活，过拟合的可能性就越大，但本书中不考虑该问题。）

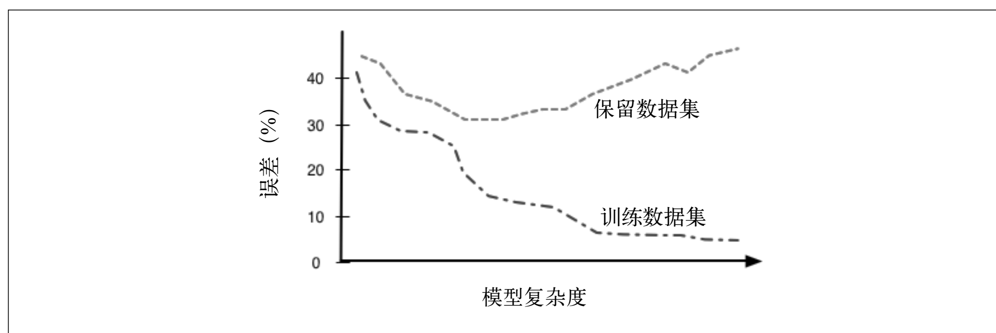


图 5-1：典型的拟合图。曲线上的点代表模型在特定复杂度下（横坐标）的预测准确率。训练集和测试集上的预测准确率随模型复杂度的不同而变化。当模型复杂度较低时，准确率不高；当模型过于复杂时，模型在训练集上的准确率会非常高。但这实际上是过拟合，该准确率会与保留（泛化）准确率截然不同

图 5-2 是前文中提到的用户流失示例中的“表模型”的拟合图。由于这个示例的极端性，这幅拟合图也相对特殊。同样，x 轴表示的是模型的复杂度（在本例中即表中的行数），而 y 轴表示的是错误率。随着表格规模的增大，表模型记住的训练数据越来越多，每增加一行新数据，训练集的错误率就随之降低。最终表格将大到包含整个训练集（x 轴上的 N 点），此时错误率将降为 0。然而，测试集（保留数据）的错误率从开始就一直是某个值（记为 b），并且从来没有下降过，这是因为训练数据集和保留数据集是没有交集的。而这两个数据集错误率的巨大差异，表明模型确实记住了训练数据。

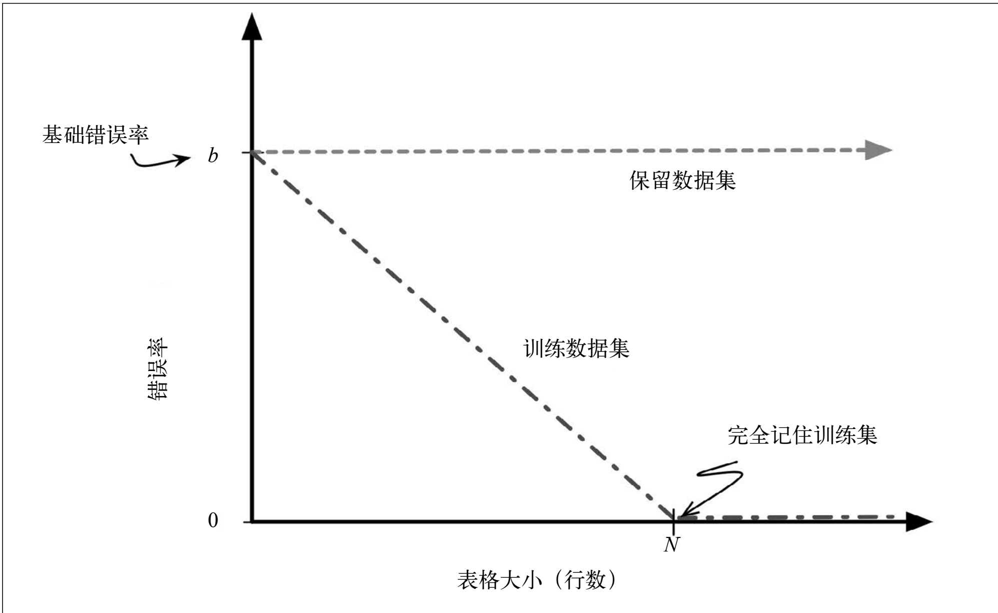


图 5-2：用户流失（表）模型的拟合图



基础比率

b 到底是什么呢？由于表模型会把每一个新案例预测为“不流失”，所以它对所有“不流失”用户的预测都是正确的，对所有“流失”用户的预测都是错误的。错误率就是这些“流失”用户在总体中所占的比例。这也称作“基础比率”。而永远选择多数类的分类器也被称作“基础比率分类器”。

回归模型的对应基线则是一个总是能预测出目标变量值的平均数或中位数的简单模型。

你偶尔也会听到“基础比率表现”这样的说法，而以上就是它所指的内容。下一章会再次回顾基础比率的概念。

由于前文中已经讨论了两种截然不同的建模过程，即反复对数据进行划分（如树型归纳）和通过找到一系列最佳参数（如线性模型的权重）来拟合数值模型，因而现在我们可以检验两者过拟合的情况了。

5.3.2 树型归纳的过拟合问题

回顾一下在解决分类问题时，构建树形结构模型的方法。在因为被反复划分而越来越小的数据子集中，我们用基本能力来找出重要的、预测能力强的单个属性。为了便于说明，假设数据集中不存在特征向量相同而目标变量值不同的两个实例。如果不断划分数据，那么最终所有的子集都将是纯集，即任意一个子集里的所有实例的目标变量值都相同。这些子集就是树上的叶节点。叶节点中可能会有多个实例，这些实例的目标变量值都相同。如果有必要，可以继续按属性划分数据集，直到每个叶节点上只有一个实例为止，而这就是“纯”。

我们刚刚做了什么？我们其实构建了一个前几节中作为过拟合极端例子提到过的查找表！每个输入树形模型进行分类的训练集中的实例都会自主选择分支，最终到达属于自己的叶节点，而该叶节点则对应着包含该实例的子集。那么这棵树在训练集上的准确率如何呢？答案是完全精确，对每一个训练集实例它都会做出正确的分类预测。

该模型能否泛化？或许吧。该模型应该比查找表略好一些，因为每个新实例都能被分到**某类**，而不是只得到一个不匹配的结果。即使是对以前没有出现过的实例，模型也会给出重要的分类。因此，凭经验来检验模型在训练集和测试集上的准确率，是很有用的。

在树形结构模型中不断分支，直到得到纯叶节点的过程很容易导致过拟合。树形结构模型在代表对象方面非常灵活，事实上，它可以代表任何特征函数，如果无限制地分支下去，那么它的准确率甚至可以达到任意水平。但这样的话，这棵树可能会非常地庞大，而树的复杂度与节点数密切相关。

图 5-3 是树型归纳的一幅典型拟合图。在这里我们人为限制了每棵树的最大规模，并通过 x 轴来衡量所限定的节点数（方便起见，将其对数化）。为了代表每棵树的规模，我们用训练集重新构造一个树形模型，并计算两个值：模型在训练数据集上的准确率和在保留数据集（测试集）上的准确率。如果叶节点上的子集不纯，那么可以根据子集中目标值的平均值来预测目标变量，正如第 3 章中讨论的那样。

这棵树起初（图像左侧）很小，预测效果也很差。随着树的节点增多，训练数据集上和保留数据集上的准确率也随之快速提高。可以看到，训练数据集上的准确率总是比保留数据集上的准确率高一些，这是因为我们在建模时使用的是训练集。但是从某一点开始，这棵树就出现了过拟合现象，如保留数据集的曲线所示，这是因为模型把训练数据集中的某些细节包括了进去，而这些细节在总体中不是普遍存在的。在本例中，过拟合现象大约出现在 $x = 100$（节点数）处，也就是图中标出的“甜蜜点”。该点之后，随着叶节点上的子集越来越小，模型的泛化能力也越来越差，因而越来越容易出现错误，同时，模型在保留数据集上的预测能力也变差。

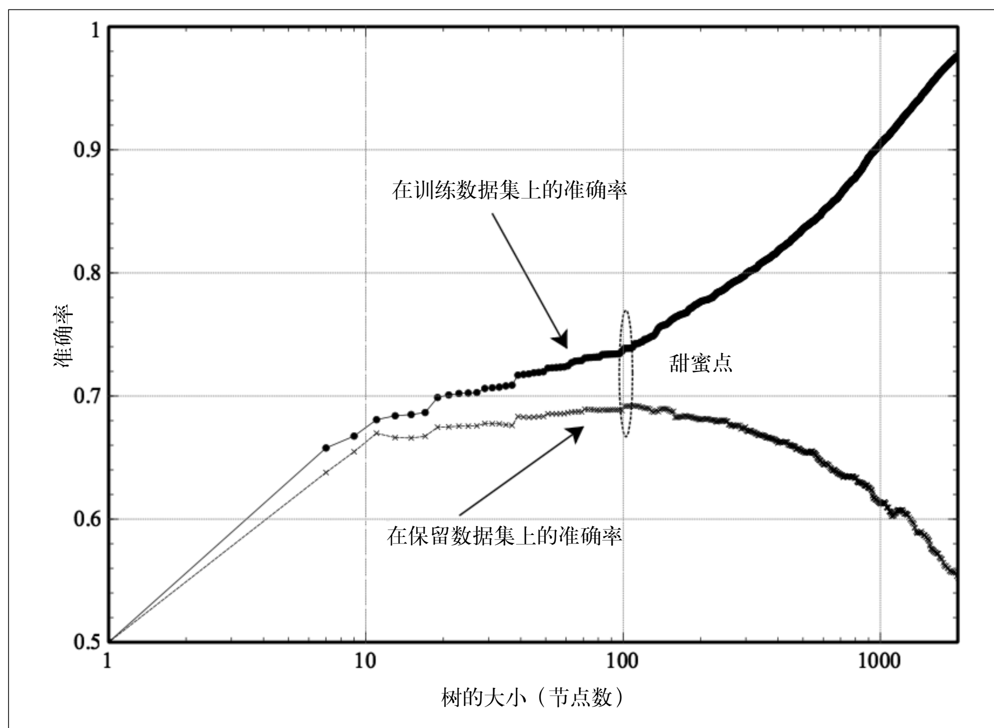


图 5-3：树型归纳的典型拟合图

总之，根据拟合图得知，数据集的过拟合现象大约出现在节点数为 100 处，因此我们应该把树的规模限制到这个值[2]。这代表了它在两个极端——一是根本不划分数据，只简单地使用整个数据集中的平均目标值；二是构建一棵完整的树，直到得到纯叶节点——之间的最佳权衡。

然而，目前还没有一种能在理论上确定甜蜜点确切位置的方法，因此我们还是需要用基于实验的技术来判断。在探讨这些实验方法之前，我们先检验一下第二种建模过程的过拟合情况。

5.3.3　数值函数的过拟合问题

控制数值函数复杂度的方法有多种，许多书都专门了讨论这个话题。本节讨论最重要的一种方法，5.9.3 节会讨论另一种。建议读者至少略读高级内容（带星号），因为其中介绍了现阶段数据科学家普遍使用，而非数据科学家却一头雾水的概念和术语。本节会对这些概念和术语进行总结，使读者能在概念层面上充分理解它们。[3] 但是首先，本节要讨论一种把

注 2：注意，100 个节点不是一个普遍适用的值，而是仅适用于这个数据集的值。如果我们对数据做出了较大改变，或仅仅改变了建树算法，就可能需要重新画一幅拟合图来寻找新的甜蜜点。

注 3：本节也会提供足够的概念工具，来帮助读者更好地理解支持向量机——它在复杂度（过拟合）控制方面与逻辑回归几乎相同。

函数变复杂的更直接的方法。

这个方法就是在函数中加入更多变量（或称属性）。举个例子，假设有一个线性模型，如公式 4-2 所示：

$$f(\boldsymbol{x}) = w_0 + w_1x_1 + w_2x_2 + w_3x_3$$

随着 x_i 的增多，函数也会变得愈加复杂。每个 x_i 都有一个对应的 w_i，即模型的学习参数。

有时建模人员还会通过加入原属性的非线性变形破坏方程的线性性质。比如，我们可以加入第四个属性 $x_4 = x_1^2$，如果觉得 x_2 和 x_3 的比值很重要，那么还可以加入第五个属性 $x_5 = x_2/x_3$。现在我们需要找到以下几个属性的参数（权重）：

$$f(\boldsymbol{x}) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + w_5x_5$$

无论用哪种方式，数据集最终都会包含大量的属性，而使用所有的属性可以给模型很大的余地来适应训练集。你可以回忆一下几何学，二维象限中的任意两点决定一条直线，三维象限中的任意三点可以决定一个平面。这一概念可以推广为：随着维度的增加，我们可以用更多的任意点来拟合更大的数据集，即使不能完美拟合，也能通过增加维度数（即属性数）来改善拟合效果。

通常，为了避免出现过拟合，建模人员会仔细修剪模型的属性。他们会用上面介绍的保留技术，对单个属性的信息进行评估。如果我们有大量人力资源，而属性又相对较少，那么对属性进行手动挑选也是一个不错的选择。但是在现在的很多应用场景中，会自动生成大量模型，而其中属性的数目也非常多，这时手动选样属性就不太合适了。比如，依赖数据科学进行线上广告精准投放的公司每周可以构造上千个模型，其中又可能包含数百万个属性，这时就只能自动选择属性了（或者干脆不进行属性选择）。

5.4　示例：线性函数的过拟合

4.1.3 节引入了一个简单的鸢尾花数据集，其中包含两种鸢尾花的描述性数据。请回顾这一示例，并考虑其中的过拟合问题。

图 5-4 把原始鸢尾花数据集中的两个属性——花瓣宽度和萼片宽度——分别作为两个坐标轴，图中每个点都代表一种鸢尾花，实心点是**山鸢尾**，圆圈是**变色鸢尾**。注意以下几点。首先，两种鸢尾花截然不同，容易区分。实际上，图中的两“簇”鸢尾花数据点中间有一道极宽的间隙。其次，逻辑回归和支持向量机都对该数据集进行了划分，但由于两条分割线非常相似，所以在图中无法分别呈现。

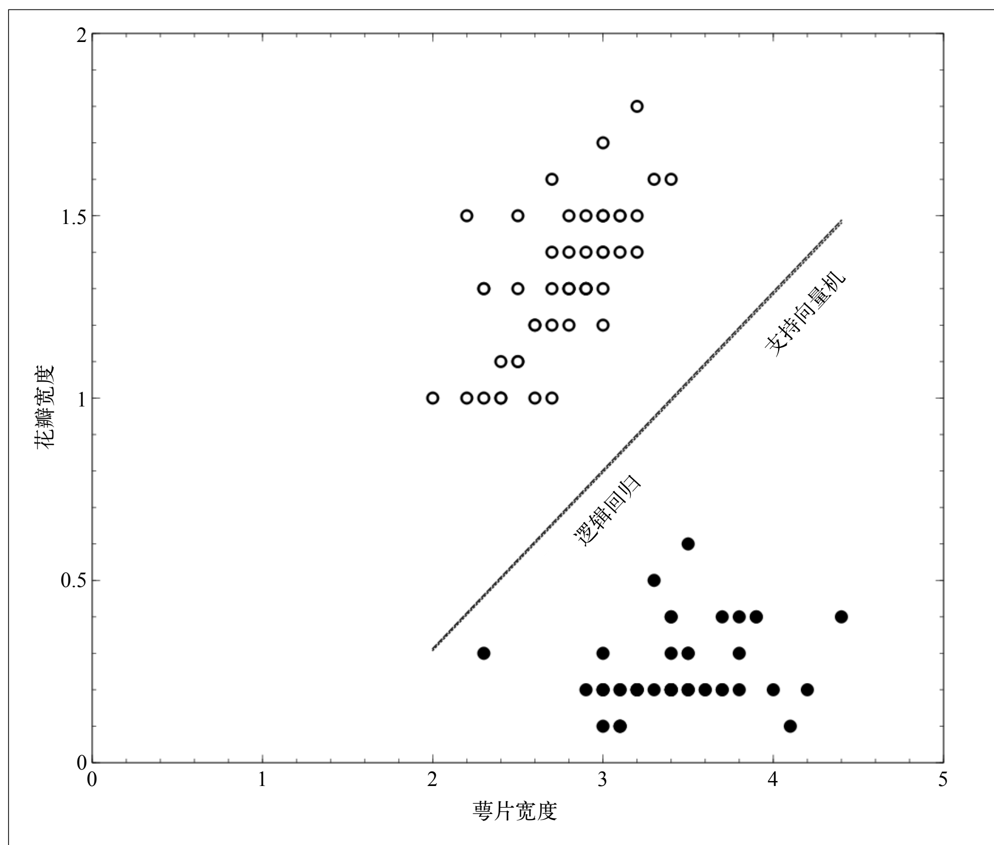


图 5-4：原始鸢尾花数据集及两种线性方法学习得到的模型（边界线）。本例中，线性回归和支持向量机学习得到了相同的模型（即图中的决策边界线）

图 5-5 中加入了坐标为 (3, 1) 的新**山鸢尾**实例。从现实角度出发，我们可能会把该点当作离群点或错误，因为比起**山鸢尾**，它更接近**变色鸢尾**的数据点群。你会发现逻辑回归线做出了相应的调整，因此依然能对两类进行完美分类，而支持向量机的线却几乎没有移动。

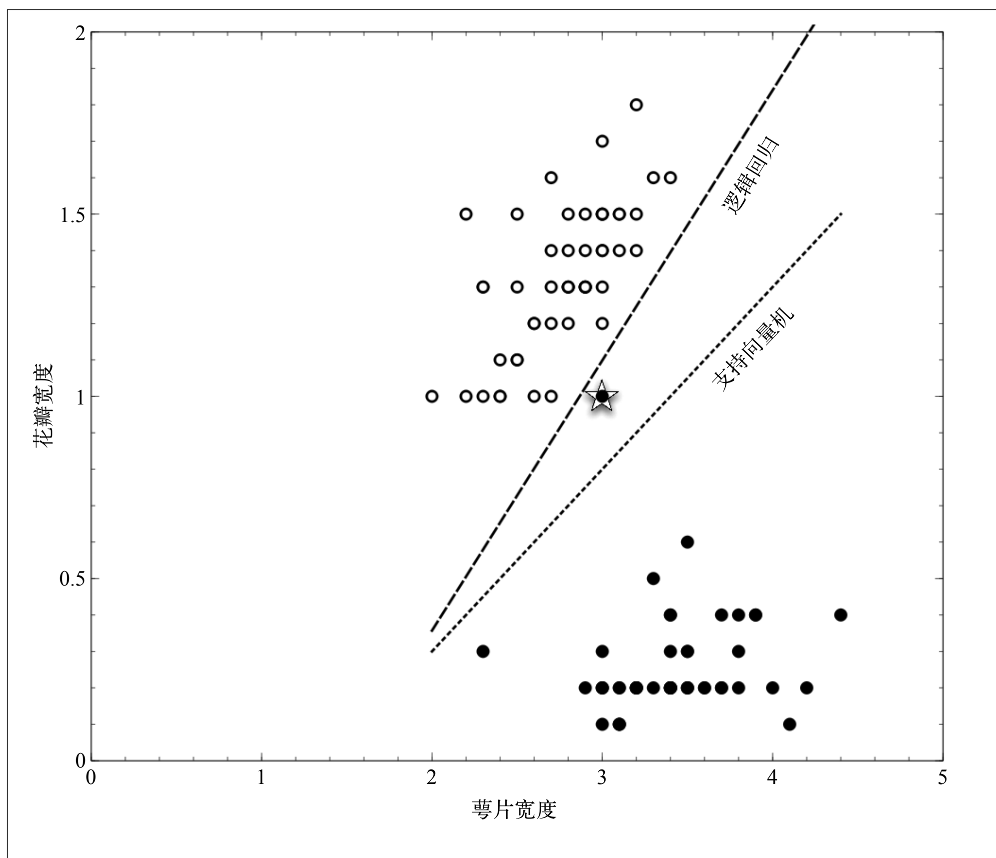


图 5-5：在图 5-4 的鸢尾花数据集中加入一个新的山鸢尾数据点（星形）。注意，逻辑回归的模型发生了很大改变

我们在图 5-6 中加入了另一个离群点 (4, 0.7)，这是一个混入**山鸢尾**数据点群的**变色鸢尾**数据点。同样，支持向量机的线几乎没有移动，而逻辑回归线的位置发生了巨大变化。

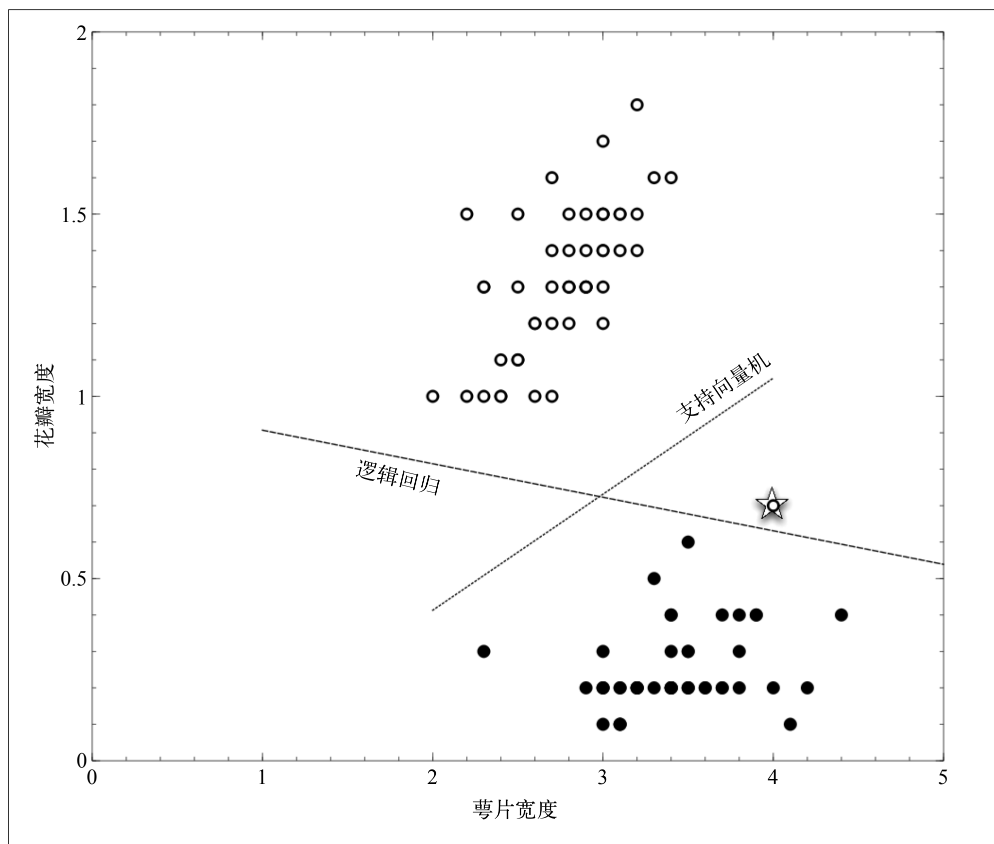


图 5-6：在图 5-4 的鸢尾花数据集中加入一个新变色鸢尾数据点（星形）。注意，逻辑回归的模型再度发生了巨大改变

图 5-5 和图 5-6 中的逻辑回归似乎都出现了过拟合问题。可以说，这两次加入的数据点都是离群点，不应该对模型产生很大的影响——它们对物种“质量”的贡献很小。但本例中逻辑回归的确受到了这种影响。只要线性边界存在，逻辑回归就能找到它[4]，哪怕这意味着需要为适应离群点而调整边界。支持向量机则不像逻辑回归那样对单个数据点如此敏感。支持向量机的训练过程包含了复杂度控制，后面会详细讨论这项技术。

注 4：严格来讲，只有一部分逻辑回归算法能确保找到线性边界，有的则不能保证。但是，这与我们在此处添加的过拟合点无关。

前文提到，另一种使数值函数变复杂的方法是加入更多变量。如图 5-7 所示，本例依然使用图 5-6 中的数据集，但加入了一个新属性——萼片宽度的平方。加入这个属性可以使模型在拟合数据时更加灵活，因为我们可以对平方项分配权重。从几何角度看，这意味着决策边界不仅可以是一条直线，还可以是一条**抛物线**，新加入的属性使得两种方法都能绘制出更贴合分布区域的曲面。当不得不使用曲线（或曲面）来进行拟合时，必须有额外的自由度，而这也使过拟合的可能性变大了。注意，无论支持向量机如何变化，即使现在其边界变成了曲线，其训练过程的本质仍是选择边界附近的最大间距，而不是对不同的正向类进行完美划分。

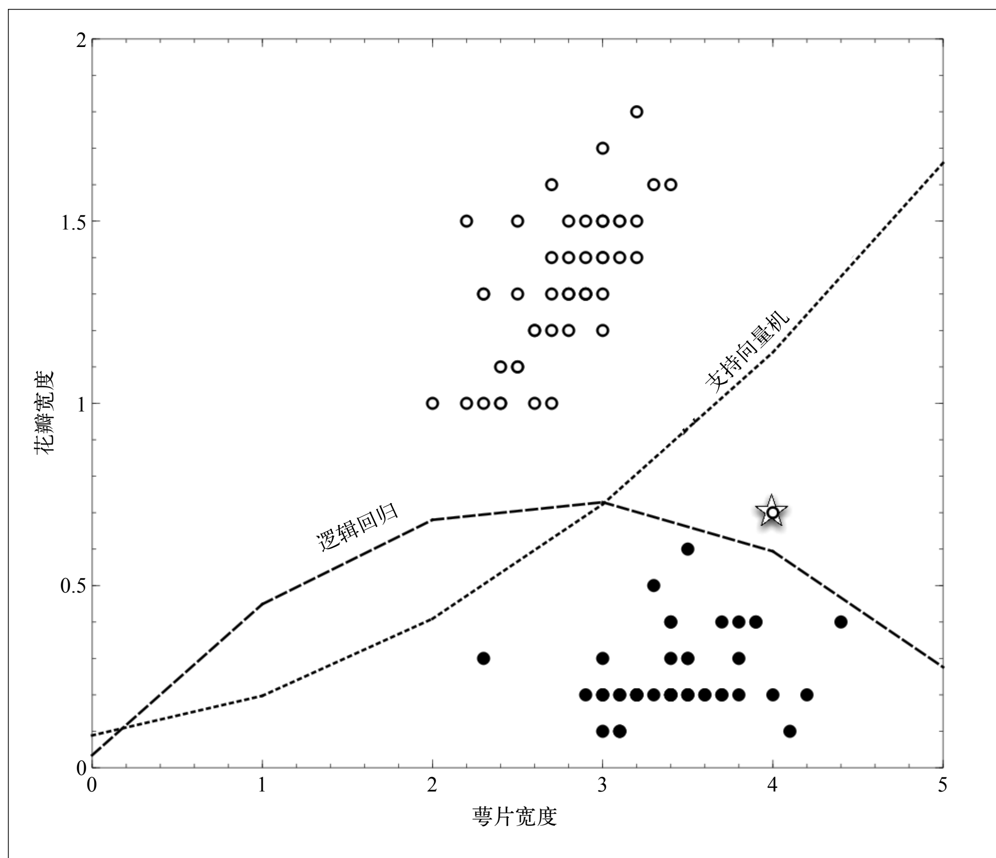


图 5-7：在图 5-6 加入了新变色鸢尾数据点（星形）的鸢尾花数据集的基础上，给逻辑回归和支持向量机各加入一个特征——萼片宽度的平方，使得两者能构造更复杂的非线性模型（边界）

5.5 *示例：过拟合为何有害



前方有技术细节！

本章开头说过，只会记忆的模型毫无用处，因为它不仅总是会过拟合，还无法泛化，但这在技术上只证明了在达到特定复杂度之后，过拟合会成为模型优化的阻碍，而并未解释过拟合为何会让模型**越来越差**（如图 5-3 所示）。本节将通过一个示例详细解释该现象产生的过程及原因，但跳过本节也不影响阅读。

为什么模型效果会越来越差呢？简单地说，随着模型复杂度的上升，模型会出现有害的虚假相关关系。而这些相关关系仅适用于建模所用的特定数据集，在总体中并不是普遍存在的。当这些虚假相关在模型中进行了**不正确的**泛化时，过拟合情况就会出现，模型效果也会变差。本节将用一个示例详细探讨这种现象发生的原因。

考虑一个简单的二分类问题，类别为 c_1 和 c_2，属性为 x 和 y。有一个实例总体，其中两类实例各占一半。属性 x 有两种取值，p 和 q；属性 y 也有两种取值，r 和 s。在总体中，$x = p$ 在 c_1 中占 75%，在 c_2 中占 25%，因此 x 能对类别进行预测。我们故意让 y 没有预测能力，而在数据样本中，y 值在两类里出现的频率也很平均。简单地说，我们很难对这些数据进行划分，因为仅有一个变量 x 能对类别进行预测，而根据 x 进行预测，所能达到的最高准确率是 75%。

表 5-1 展示了实例总体中一个很小的训练数据集。如何根据这些数据构建分类树呢？此处不展开熵值计算，你只需知道属性 x 可以产生某种影响，我们可以据此对树进行分支，生成如图 5-8 所示的树。由于只有 x 能够预测目标变量，所以这棵树就是最优树了。它的错误率为 25%，相当于理论上的最小错误率。

表5-1：一个小型训练样本

实例	x	y	类别
1	p	r	c_1
2	p	r	c_1
3	p	r	c_1
4	q	s	c_1
5	p	s	c_2
6	q	r	c_2
7	q	s	c_2
8	q	r	c_2

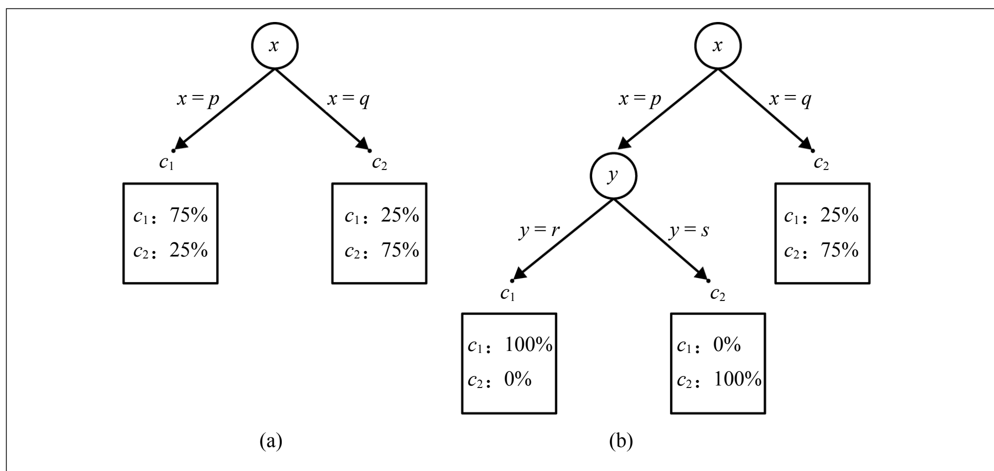


图 5-8：过拟合示例的分类树。(a) 最优树仅有 3 个节点；(b) 过拟合的树能更好地拟合训练集数据，但泛化准确率较差，因为其外部结构无法做出最佳预测

然而，表 5-1 中 y 的两个值 r 和 s 并没有均匀分布在两类中，因此 y 似乎也能对目标变量值进行预测。尤其当我们选择 $x = p$（实例 1~4）时，可以看出 $y = r$ 能完美预测出分类 c_1（实例 1~3）。因此对于该数据集而言，我们可以通过按 y 值划分生成两个新叶节点来获得信息增益。

该训练集显示，树 (b) 比树 (a) 表现得要好。前者预测正确了 8 个训练样本中的 7 个，而后者只预测正确了 6 个。但这是因为数据样本中 $y = r$ 恰巧与类 c_1 相关，而在整个总体中并不存在这样的相关性。树 (b) 的这个多余分支误导了我们，它不仅无用，甚至还有害。回忆一下，整个总体中 75% 的 $x = p$ 的实例分布在 c_1 中，25% 分布在 c_2 中。但是，树中的假分支 $y = s$ 对类别 c_2 的预测在整个总体中是不正确的。实际上，我们估计这个假分支导致的误差要占这棵树总误差的八分之一。总的来说，树 (b) 的总期望误差为 30%，而树 (a) 仅为 25%。

最后，要强调几点。首先，这种现象不只出现在分类树中。我们选择树形模型是为了便于指出假分支，但所有模型都容易受过拟合的影响。其次，这种现象的出现不是因为表 5-1 中的训练数据不典型或有偏，每个数据集都是更大的总体的一部分，即使抽样无偏，样本也会存在波动。最后，正如前文所说，能事先判断模型是否出现过拟合的一般性分析方法不存在。在本例中，我们事先知道总体特征，因此可以判断模型是否出现了过拟合，然而在现实中，你不会事先得到这样的信息，因此必须用保留数据集来检测过拟合现象。

5.6 从保留评估到交叉验证

本书稍后会展示一种用来避免过拟合的应用广泛的通用技术，这种技术能够应用于属性选择和树的复杂度等问题。但我们首先需要详细探讨一下保留评估。在避免过拟合之前，我们需要先注意别被过拟合骗了。本章开头引入了一个概念，即为了对模型的泛化能力有一个公允的评估，应该先估计这个模型在保留数据集（未用来建模，但目标变量值已知的数据集）上的准确率。保留评估往往与其他“实验室”情境下的评估非常类似。

即使保留数据集的确能给出泛化能力的估计，这也只是一种单一估计罢了。我们能信任这种准确率的单一估计吗？最终得到这个估计结果，可能只是因为幸运地选对了（或不幸地选错了）训练集和测试集。本章不会详细讨论计算置信区间的方法，但讨论其一般的测试过程还是非常重要的，因为这在很多方面都很有帮助。

交叉验证是一种更为复杂的保留训练和保留测试过程。我们不仅想要对泛化能力的简单估计，还想要所估计出的泛化能力的一些统计数据，如均值和方差，以便了解该泛化能力在多个数据集之间的变化。你可能已经在统计课上学过，方差是评估能力估计值的置信度的关键指标。

交叉验证也能使有限数据集发挥更大的作用。交叉验证不是将数据集拆分成一个训练集和一个测试集，而是通过反复划分并系统地交换训练集与测试集，计算所有数据组合的估计值。

构建模型“实验室”

建造建模实验室的基础设施可能价格不菲且耗费时间，但在投资之后，模型许多方面的性能都能在可控的情况下进行快速评估。然而，因为保留检验并不能反映模型在现实世界中遇到的所有复杂因素，所以数据科学家需要努力理解实际应用场景，并尽可能地把实验室配置得与之类似，以防与现实差异太大。举个例子。一家公司想用数据科学来为价格昂贵的个人定向广告进行精准投放。随着活动的进行，该公司收集到了越来越多的关于客户收看广告后是否购买的数据，这些数据就可以用来建立模型，来区分应该投放广告的人和不应该投放广告的人。我们把这个例子放在一边，先来评估一下预测客户是否会对广告做出响应的模型的准确率。

当模型被投入使用，作用在那些“自然”的客户后，公司会惊讶地发现，这些模型并没有在实验室中表现得那么好。这是为什么呢？虽然原因有很多，但最重要的一个是：训练数据和保留数据与模型接触的实际数据并不相符。尤其是，训练数据中都是已经被精准投放过该广告的客户，而在现实生活中，我们并不知道客户的目标变量值（是否做出响应）。即使在使用数据挖掘之前，公司也不是简单地任意确定目标，而是根据某些标准把他们认为会做出响应的客户作为目标。在实际应用中，模型面对的是更为广泛的客户群，而不仅仅是符合标准的客户。训练数据和实际数据的差异可能是模型效果退化的原因。

这种现象不仅出现在广告精准投放中。考虑一下信用评分问题。我们想建立一个能预测客户违约概率的模型，而数据中的“有不良贷款”和“无不良贷款”两类都基于曾经被发放过贷款的客户，即我们认为违约风险较低的客户。

在这两种情况下，请你考虑如何找到更恰当的数据集来构建预测模型。别忘了应用第1章中的基本概念：把数据当成你要**投资**的资产。

交叉验证一开始会把标签数据集划分成 k 个子集，这些子集被称为**折叠**（fold）。一般情况下，k 等于 5 或 10。图 5-9 的上半部分就是一个被分为 5 个折叠的标签数据集（原始数据集）。随后我们用交叉验证，以一种特殊的方式将训练过程和测试过程进行 k 次迭代。如

图 5-9 的下半部分所示，在每一次交叉验证迭代中，都有一个折叠被选作测试集，而其他 k–1 个折叠则被整合起来作为训练集，因此，在每次迭代中，有 (k–1)/k 的数据作为训练集，1/k 的数据作为测试集。

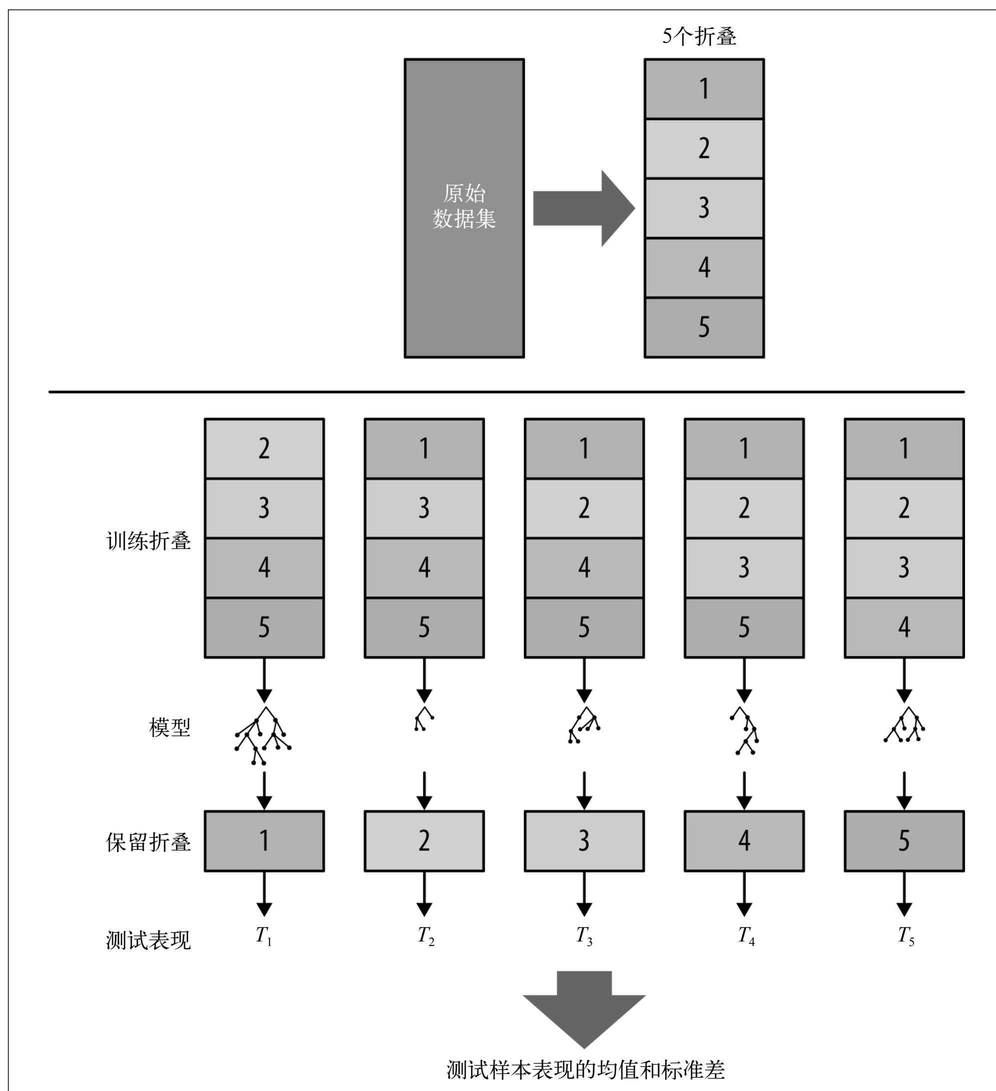


图 5-9：交叉验证图解。交叉验证的目的是有效利用标签数据来估计建模的效果。此处展示了一个 5 重交叉验证，原始数据集被随机分为 5 个等规模的子集，随即每个子集轮流作为测试集，而其余 4 个则作为训练集。最终得到 5 个不同的准确率，我们可以计算其均值和方差

每次迭代可以得到一个模型，从而得到一个泛化能力的估计指标，如准确率估计。交叉验证结束时，每个子集都会有一次被作为测试集，有 k–1 次作为训练集。此时我们有了所有 k 个折叠的效果估计，从而能够计算均值和标准差。

5.7 用户流失数据集回顾

回想 3.6 节中的用户流失数据集。在那一节中，整个数据集既被用作训练集，又被用作测试集，最终的准确率为 73%。该节末尾提了一个问题：**你相信这个数字吗**？此刻你应该已经明白，不应该相信任何用训练集做测试得到的准确率，因为过拟合的可能性太大了。既然我们学习了交叉验证，就不妨重新仔细地进行一次评估。

图 5-10 展示了 10 重交叉验证的结果。图中其实存在两种模型，上半部分表示逻辑回归的结果，下半部分则表示分类树的结果。更确切地说，我们打乱了数据集，将其划分为 10 等份，而这 10 等份轮流作为保留数据集，另外 9 份合起来作为训练集。各部分中的水平线代表该类的 10 个模型的平均准确率。

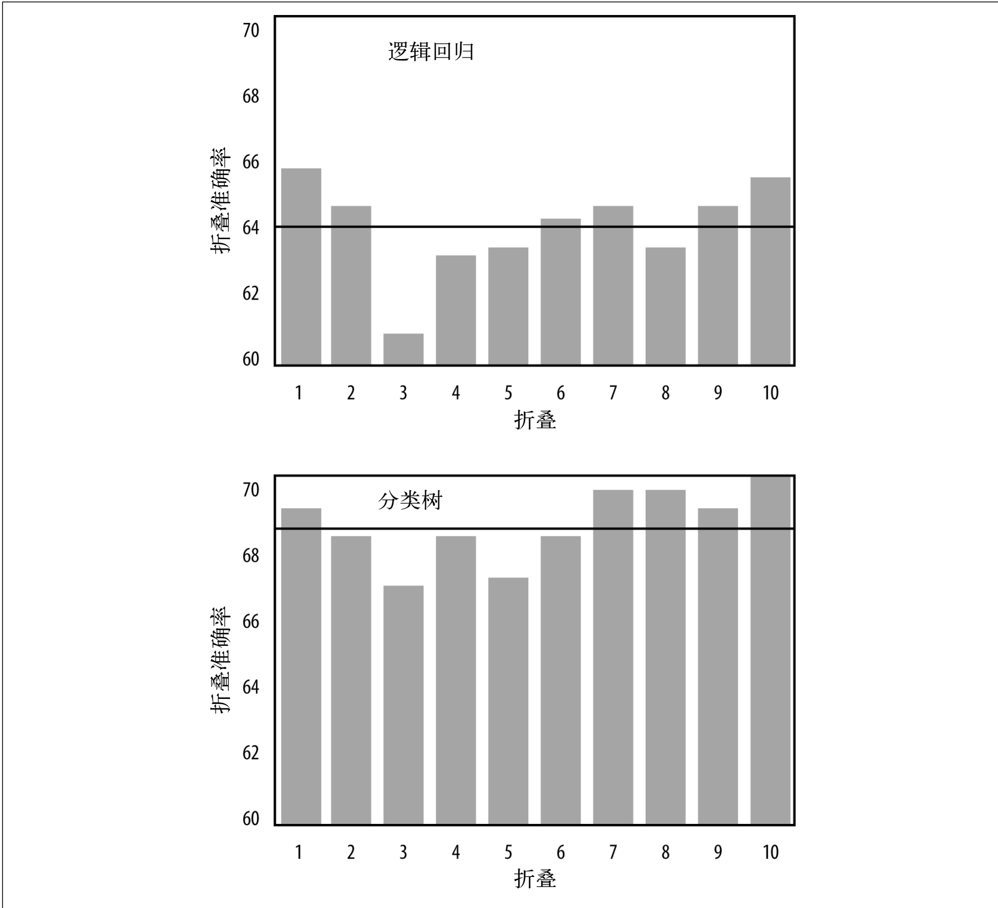


图 5-10：用户流失问题交叉验证的折叠准确率。上半部分是逻辑回归的准确率，我们把含有 20 000 个实例的数据集等分成 10 个折叠。下半部分则代表了对同样的折叠所做的分类树的准确率统计。每幅图中的水平线代表折叠的平均准确率（注意 *y* 轴值域的选择，这突出了准确率的差异）

我们可以从图中得到许多信息。首先，经过交叉验证得出分类树的平均准确率是 68.6%，明显比之前计算出的 73% 要低，这意味着分类树中的确存在过拟合现象，且新数值（较小值）更可信。其次，在不同折叠情况下，模型的效果存在差异（折叠准确率的标准差是 1.1），因此最好既对它们取平均来大体了解模型的效果，也求出预期分类树在这个数据集上的方差。

最后，通过比较逻辑回归和分类树的折叠准确率可知，两部分存在一些共性，比如，两种模型在第 3 个折叠上的效果都不佳，而在第 10 个折叠上的效果都较好。但两者仍是截然不同的。值得注意的是，与分类树相比，逻辑回归的平均准确率略低（64.1%），而方差更高（标准差为 1.3）。对这个数据集而言，树形模型可能比逻辑回归更适用，因为前者更稳定、效果更好。但这个结论不是绝对的，我们会也看到，在其他数据集上其结果可能相反。

5.8 学习曲线

如果训练集的大小改变，那么你可能觉得从中得出的模型的泛化能力也会改变。若其他因素不变，在一定程度上，训练集数据越多，模型的泛化能力就越强。描绘模型泛化能力与训练集数据量关系的图线叫作**学习曲线**，这也是一种重要的分析工具。

图 5-11 展示了电信公司用户流失问题的树型归纳和逻辑回归的学习曲线。[5] 学习曲线的形状很有特色。起初，建模程序找到数据集中最明显的规律时，曲线会较为陡峭。然后，随着训练集的规模增大，更精确的模型出现了，但因为数据量增加带来的边际收益降低了，所以学习曲线的陡峭程度也降低了。有时候，曲线会完全变平，因为即使训练集再增大，模型的准确率也不会上升了。

理解学习曲线和拟合图（或称拟合曲线）之间的区别很重要。学习曲线展示了随着所使用的训练数据量的变化，模型泛化能力（仅在测试集上）的变化：而拟合图则展示了随着模型复杂度的变化，其泛化能力的变化和该能力在训练集上的变化，与模型**复杂度**的曲线。拟合图通常用于展示数据量固定的训练集。

即使数据相同，不同的建模过程输出的学习曲线也可能千差万别。从图 5-11 中可以看到，当训练集较小时，逻辑回归的泛化能力强于树型归纳。然而，当训练集数据量增大时，逻辑回归的学习曲线则更快趋于平稳，两条曲线交叉，树型归纳随即占了上风。这种现象与之前所说的“模型越灵活，过拟合越严重”有关。在特征相同的情况下，分类树比线性逻辑回归更为灵活，这会导致两种结果：一是数据集较小时，树型归纳的过拟合情况可能更为严重，我们常看到如图 5-11 所示的情况，即逻辑回归在小数据集上的效果更好（但不总是这样）；二是图中同样显示，树型归纳的灵活性使其在大训练集上更占优势，因为树能代表特征变量与目标变量之间的大量非线性关系。至于树型归纳对这些关系的反映是否真实，需要我们基于经验进行分析，即应用诸如学习曲线一类的分析工具。

注 5：Perlich 等人（2003）展示了树型归纳和逻辑回归的学习曲线，以解决多个分类问题。

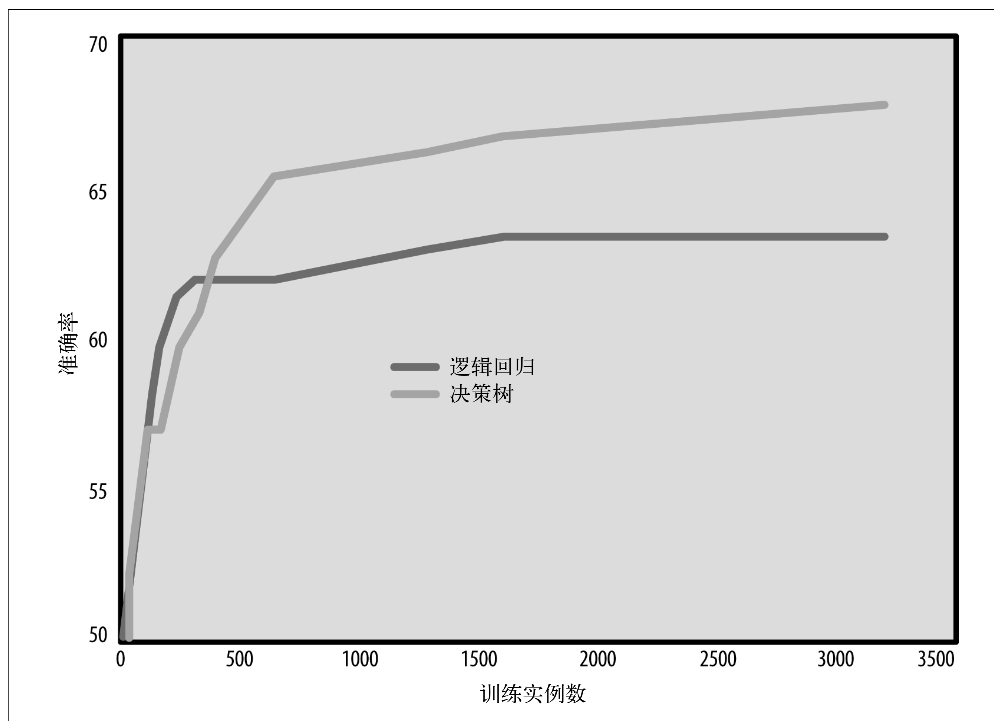


图 5-11：用户流失问题的树型归纳和逻辑回归学习曲线。随着训练集数据量（*x* 轴）的增加，泛化能力（*y* 轴）逐渐增强。重要的是，两种技术随训练集数据量的增长而产生的提升率不同，提升率的变化也不同。因为逻辑回归灵活性较低，所以在小数据集上过拟合情况较轻，却不能完全反映数据的复杂度；而因为树型归纳更为灵活，所以虽然它在小数据集上过拟合情况较为严重，却能反映大数据集的复杂规律



学习曲线还有其他分析用途。比如，本书已经指出，数据可以作为一项资产，而学习曲线可以告诉我们：泛化能力已经趋平，花费成本获取更多训练数据是不值得的。这种情况下，我们应该接受当前的泛化能力水平，或是寻找其他改进模型的方法，比如设计更好的特征。学习曲线还可以告诉我们：泛化准确率正持续上升，获得更多的训练数据可能是一项很好的投资。

5.9 避免过拟合与控制复杂度

为了避免过拟合，需要控制从数据中得出的模型的复杂度。我们先来看看树型归纳的复杂度控制。因为树型归纳较为灵活，如果不引入某种机制进行控制，那么其过拟合可能会很严重。而通过在树型归纳背景下的讨论，我们可以获得一个同样适用于其他模型的一般性机制。

5.9.1　树型归纳中的过拟合规避

树型归纳的主要问题是，模型会持续分裂，直到得到纯的叶节点。这会使树变得非常庞大非常复杂，且出现过拟合，而我们已经知道了这种问题的致命性。树型归纳通常使用两种规避过拟合的技术，一是停止分裂以防树过于复杂，二是先让树持续分裂，直至其过于庞大，然后剪枝，缩小其规模（降低其复杂度）。

有许多方法可以达到这两个目的。限制树规模的最简单方法是规定叶节点上实例数的最小值。“实例数最小值”停止准则背后的思想是：在预测建模中，我们要用叶节点上的数据，对未来落在该节点上的实例的目标变量进行统计估计。如果用于估计的基础数据量非常小，那么估计的准确率可能会比较低，在我们试图得到纯叶节点时尤其如此。用这种方法控制复杂度的一个好处是，树型归纳会自动对数据量较大的节点进行分支，并剪除数据量较小的分支，从而基于数据分布自动调整模型。

关键问题是，该如何选择阈值，即叶节点上至少要有几个实例？ 5 个？ 30 个？还是 100 个？尽管内行人士通常会根据经验做出自己的选择，但放之四海而皆准的数字其实是不存在的。然而，研究人员已经开发出了用统计方式判断停止点的技术。你也许在基础统计课上学过统计学中“假设检验”的概念。大致上，假设检验可以用于检验某些统计数据中的差异是否为偶然现象。统计假设在大多数情况下依赖“*p* 值”，该值可以用于判断统计中的差异出于偶然的概率，若其低于某个阈值（通常是 5%，但也因问题而异），假设检验就会判断该差异并非出于偶然。因此，另一种限制叶节点上实例数从而使树停止分裂的方法，就是对每个叶节点进行假设检验，判断信息增益的差异是否出于偶然。如果假设检验判断并非如此，那么树就继续分裂。（见 5.9.3 节补充栏。）

另一种降低过拟合程度的方法是对过大的树进行“剪枝”，即用叶节点代替原有的叶节点和分支。剪枝的方法有很多，感兴趣的读者可以在数据挖掘文献中获取更多细节。该方法的总体思路是判断用叶节点代替一系列叶节点和分支后，准确率是否会下降。如果没有，则进行剪枝。我们可以在子树上不断迭代这个过程，直到删除或代替任何分支都会使准确率下降为止。

最后，提供一种能够推广到不同建模过程中的方法。考虑一个问题：如果构造出复杂度不同的树怎么办？比如，我们在有一个节点之后就停止建树，再构造一个含两个节点的树，然后是三个节点……从而得到一组复杂度各异的树。如果有估计它们的泛化能力的方法，就可以选择（估计）泛化能力最佳的那棵树了！

5.9.2　避免过拟合的一般方法

一般来说，如果有一系列复杂度各异的模型，那么通过估计它们的泛化能力，就能选出最佳模型。但是如何估计它们的泛化能力呢？用（带标签的）测试集吗？这存在一个严重的问题：测试集必须严格**独立**于模型，这样才能得到模型准确率的独立估计。比如，我们可能想估计最终的企业绩效，或比较分别由两种方法（如分类树和逻辑回归）构建的最佳模型。如果不想比较模型或得到模型准确率和（或）方差的独立估计，那么可以仅仅根据测试数据选出最佳模型。

但是，即使我们想要这些东西，也可以继续。关键是要意识到，第一次训练 / 测试划分并没有什么特别之处。假设要把测试集留作最后评估用，就可以把训练集继续划分成训练子

集和测试子集，然后用这个新的训练子集构建模型，而用新的测试子集挑选最佳模型。为了避免混淆，我们称前者为**子训练集**，称后者为**验证集**。验证集与最后的测试集分开，后者不会用于做出任何建模决策。该过程通常被称作**嵌套保留测试**。

回到分类树示例，我们可以由子训练集得到许多复杂度各异的树，然后用验证集估计它们各自的泛化能力，最优模型对应的就是图 5-3 中倒 U 形保留曲线的顶点。假设这次评估得出的最优模型含有 122 个节点（“甜蜜点”），那么我们可以选择使用该模型，最后借助测试集来估计其实际泛化能力。我们也可以再做一步：由于我们为了选择复杂度而预留了一部分数据作为验证集，所以该模型是用原训练集的子集构造的，但既然已经选好了复杂度，那么为什么不用整个原始训练集构造一棵 122 个节点的新树呢？这样便能两全其美：既能在避免动用测试集的情况下，用子训练集和验证集得到最佳的复杂度；**又能**用整个训练集（子训练集加上验证集）构造一个最佳复杂度的模型。

许多建模算法都用以上方法来控制复杂度，通过某种嵌套保留过程来选择一些复杂度参数的值。再次声明，该方法之所以是嵌套的，是因为我们在第一次保留过程选择出的训练集上执行第二次保留过程。

我们往往还会使用嵌套交叉验证，这要更复杂些，但仍可以正常进行。假设我们想用交叉验证来评估新的建模方法的泛化准确率，该方法包含一个可调节的复杂度参数 C，而我们不知道该如何设置它。按上文所说的步骤进行交叉验证，但在对每个折叠建模之前，先用训练集（指图 5-9）做一次实验，即在该训练集上做一套完整的交叉验证，以找到最佳准确率对应的 C 值。这个实验的结果仅用于设置 C 的值，以构建交叉验证的折叠的实际模型。然后，用整个训练集构建另一个模型，其复杂度参数值为 C，并用对应的测试集来测试。嵌套交叉验证与常规交叉验证的唯一区别是，针对每个折叠，我们会先用另一个更小的交叉验证来寻找 C 值。

一旦理解了上面的解释，你就会明白，如果要在两种情况下做 5 重交叉验证，那么整个过程中实际需要构造 30 个模型（是的，**30** 个）。因为这种实验性复杂度控制建模方法计算负担过重，所以直到近几十年才得以广泛应用。

这种通过用数据进行实验来选择复杂度和构建模型的想法，适用于不同的归纳算法和不同类型的复杂度。比如，前文提过，因为复杂度会随特征集规模的增大而提高，所以往往需要精选特征集。一种常用方法是，用这种嵌套保留过程对许多不同特征集进行建模，选出最佳组合。

比如，对特征进行**序列前向选择**（SFS）。该方法通过观察所有用单独一个特征构建的模型，先用嵌套保留过程选出一个最佳的单独特征，之后再检验所有用该特征和另一个其他特征构建的模型，从中选出最佳的一个。接下来一遍遍重复该过程，选出三个特征、四个特征……直到添加特征无法提高验证集上的分类准确率为止。（与之相似的过程是**序列反向淘汰**。你应该猜到了，该过程就是从所有特征开始，一次淘汰一个。只要泛化能力不降低，该过程就一直继续。）

这是一种通用的方法。在拥有丰富数据和强大计算能力的今天，数据科学家通常会用一些战术性的嵌套保留测试（一般是嵌套交叉验证）来确定模型参数。

下一节会展示该方法在学习数值函数的过程中（如第 4 章所述）控制过拟合的另一种方式。建议读者至少略读该节，因为其中会介绍目前数据科学家常用的概念和术语。

5.9.3 *参数优化中的过拟合规避

前面讲过，避免过拟合涉及复杂度控制：在数据拟合和模型复杂性之间找到“恰当”的平衡点。我们已经学习了在用树型归纳拟合数据时如何控制树的规模（即复杂度）。而与树不同，诸如逻辑回归之类的公式，不会自动挑选属性。它们的复杂度可以通过选择一系列“恰当”的属性来控制。

第 4 章介绍了很多类现在流行的方法，这些方法通过一组可以优化拟合数据的数值参数来建模。本书到此已经探讨了其中的许多线性方法，包括线性判别式学习、线性回归和逻辑回归，许多非线性模型也能以同样的方式拟合数据。

读到这里，尤其是看过 5.4 节后，你可能会认为这些过程也会过拟合数据。然而，显式的优化框架使得它们的复杂度控制方法巧妙而富有技术性。其总体策略是，不仅要优化对数据的拟合效果，还要优化一些兼顾了拟合效果和简洁程度的组合。拟合数据的效果越好，模型就越好；同样，复杂度越低，模型也就越好。这套一般性方法叫作**正则化**，你会经常在关于数据科学的探讨中听到它。



前方有技术细节！

本节的剩余部分会（略微技术性地）简要讨论正则化的方法。别担心无法理解其中的技术细节，你只要记住，正则化不仅要优化数据的拟合，还要优化拟合的组合和模型的简易度即可。

回忆第 4 章所述，为了拟合包含数值参数 $\boldsymbol{w}$ 的模型，需要找到能将代表拟合效果的“目标函数”最大化的参数组合：

$$\underset{\boldsymbol{w}}{\arg\max}\ \text{fit}\,(\boldsymbol{x},\boldsymbol{w})$$

（$\arg\max_{\boldsymbol{w}}$ 代表你想将由所有可能的参数 $\boldsymbol{w}$ 构成的拟合最大化，且想知道能使之最大化的具体参数 $\boldsymbol{w}$。这些便是最终模型的参数。）

通过正则化控制复杂度，就是在目标函数中加入一个复杂度惩罚项：

$$\underset{\boldsymbol{w}}{\arg\max}\ [\text{fit}(\boldsymbol{x},\boldsymbol{w})-\lambda\cdot\text{penalty}(\boldsymbol{w})]$$

λ 是优化过程针对惩罚项（penalty）规定的权重（相较于拟合参数）。此时，建模人员需要决定 λ 的大小和惩罚函数。

因此，请回忆 4.3.1 节中根据数据学习标准逻辑回归模型的具体示例。我们要找到最有可能产生观测数据的线性模型——“最大似然”模型——的数值参数 $\boldsymbol{w}$。我们以下方公式表示：

$$\underset{\boldsymbol{w}}{\arg\max}\ g_{\text{likelihood}}(\boldsymbol{x},\boldsymbol{w})$$

（以上公式中“likelihood”表示“似然”，下面公式中也是如此。）而要学习一个正则化逻辑回归模型，则要计算：

$$\arg\max_{w}\,[g_{\text{likelihood}}(\boldsymbol{x},\boldsymbol{w})-\lambda\cdot\text{penalty}(\boldsymbol{w})]$$

惩罚项有多种，其性质也各不相同。[6] 最常用的惩罚项是权重平方之和，有时被称作权重 $\boldsymbol{w}$ 的“L2 范数”。该惩罚项之所以常用原因是技术性的，但基本上，如果函数包含绝对值极大的权重，则其拟合效果往往也较好。权重平方之和就能在权重绝对值极大的情况下给出较大的惩罚项。

如果在标准最小二乘线性回归中加入 L2 范数惩罚项，就能得到一个统计过程，被称为**岭回归**。如果我们选的是绝对值之和（而非其平方之和），即“L1 范数”，就能得到 **LASSO 回归**（Hastie 等，2009），该回归一般情况下被称为“L1 正则化”。出于技术性原因，L1 正则化最终会删除许多系数。因为这些系数是特征的权重之积，所以 L1 正则化能高效地自动进行特征选择。

4.1.5 节提供了详细描述线性支持向量机的机制。我们还知道，支持向量机通过拟合类之间“最宽的间隙”将“边缘最大化”。第 4 章还讨论了支持向量机用合页损失函数（见 4.2 节中的“损失函数”）来惩罚误差。现在我们可以直接把以上内容与逻辑回归结合起来。特别要注意的是，线性支持向量机学习与刚刚讨论的 L2 正则化逻辑回归几乎相同，两者唯一的区别是，支持向量机用的是合页损失函数，而不是优化的似然性。支持向量机可对以下公式进行优化：

$$\arg\max_{w}\,[-g_{\text{hinge}}(\boldsymbol{x},\boldsymbol{w})-\lambda\cdot\text{penalty}(\boldsymbol{w})]$$

其中合页损失函数项 g_{hinge} 被取消，因为合页损失越低越好。

最后，你可能会对自己说：这一切都很好，但似乎有很多神奇之处隐藏在这个 λ 参数中，而建模人员必须选择它。在诸如用户流失预测、线上广告精准投放和欺诈检测等实际问题中，建模人员该如何选择这个参数呢？

其实，我们已经有了一种直接选择 λ 的方法。前面讨论过，我们可以通过嵌套交叉验证来选择树的最佳规模和最佳特征集，其实选择 λ 的方法也是一样的。交叉验证可以在训练集子集上构建自动实验，找到最佳的 λ 值，然后用该 λ 值在所有训练数据上学习正则化模型。该过程已是构建能较好权衡数据拟合效果和模型复杂度的数值模型的标准过程。这种对数据挖掘过程中的参数值进行最优化的方法叫作网格搜索。

小心“多重比较”

试想以下情景。你开了一家投资公司。五年前，你想持有一些低市值共同基金产品，以便日后销售，但是因为你的分析师们非常不擅长挑选低市值股票，所以你采取了以下方法。先选择 1000 只不同的共同基金，每只共同基金都由罗素 2000 指数（低市值股票的主要指数）中的随机几只股票构成。你的公司秘密地对所有 1000 只基金进行了投资，并在五年后观察其收益。由于它们本身由不同的股票构成，所以它们的回报也不同，有的回报也许和指数基本相同，有的回报更低，有的则更高。最好的那只回报可能比指数高得多。现在，你卖出了大部分基金，只留下了表现最好的几只，并公开了这个消息。你可以“诚实”地说，你手上基金的五年期回报大大超过了罗素 2000 指数的回报。

注 6：《统计学习基础》里包含对这些惩罚项的绝佳技术性探讨。

这有什么问题呢？问题在于你对股票的选择是随机的！你完全不知道构成这些“最佳”基金的股票之所以表现得好，是因为它们的确优秀，还是因为你在一大堆表现各异的股票中挑选出了最好的几只。如果你抛 1000 个均匀硬币，每个抛足够多次，那么其中一个硬币抛出正面的概率可能会超过 50%。但是，选出“概率最高”的那个硬币来继续抛无疑是愚蠢的。以上都是“多重比较问题”的例子，这些问题是非常重要的统计现象，商业分析师和数据科学家必须时刻牢记。当有人在多次实验后选出最好的结果时，一定要当心。统计学教材会提醒你不要在多次统计假设实验后选出最“突出”的结果，因为这些结果通常违背了统计实验背后的假设，而结果的实际效果也令人怀疑。

模型出现过拟合其实也是因为多重比较问题（Jensen & Cohen, 2000）。注意，即使是避免过拟合的过程本身也存在多重比较（比如，通过比较选出模型的最佳复杂度）。尽管不存在什么良方能获取真正“最优”的拟合数据的模型，但我们可以通过应用本章中讨论的保留过程，以及在公布结果前仔细检查（如果可能的话），尽可能降低过拟合。比如，我们可以确信，倒 U 形的拟合图线的顶点，的确比呈任意形状的拟合图线的顶点反映的复杂度更“好”。

5.10 小结

数据挖掘包含模型复杂度和过拟合概率之间的基本权衡。如果数据所表现的现象本身就很复杂，那么就有必要构建一个复杂的模型，但复杂的模型对训练数据过拟合的风险也较高（比如模型刻画了数据总体中非典型的特征）。过拟合的模型很难适用于其他数据，哪怕这些数据都来自同一个总体。

各种类型的模型都可能出现过拟合现象。消除过拟合的万能方法是不存在的。最好的方法是通过用保留数据集进行测试来识别过拟合现象。许多曲线都有助于发现和度量过拟合现象，比如，**拟合图**中的两条曲线就以复杂度函数的形式，分别表示了模型在训练集和测试集上的效果。训练集的拟合图线通常呈 U 形或倒 U 形（取决于绘图对象是错误率还是准确率）。起初，模型非常简单，准确率也很低。随着模型复杂度的提升，准确率也会提升。随后，准确率会趋于平稳。而在过拟合情况出现时，准确率又开始下降。再比如，**学习曲线**描绘了测试集上的模型效果与所用**训练数据量**的关系，通常情况下，模型效果随数据量增大而提升，但不同模型的提升率和最终的渐近性能各不相同。

交叉验证是一种常用的实验方法，它规定了一种划分单个数据集的系统性方法。它能生成多个评估指标，而这些指标可以告诉数据科学家模型的平均水平和预期变化。

控制模型复杂度以避免过拟合的一般方法叫模型**正则化**，具体技术包括剪枝（对过大的分类树进行修剪）、特征选择，以及在用于建模的目标函数中加入显式复杂度惩罚项。

第6章

相似性、近邻和簇

基本概念：计算由数据描述的对象的相似性；运用相似性进行预测；基于相似性划分聚类

示例方法：寻找相似个体；最近邻法；聚类方法；用于计算相似性的距离度量方法

相似性是许多数据科学方法和商业问题解决方案的基础。如果说两个个体（人、企业、产品等）在某个方面是相似的，那么它们在其他方面往往也有共通之处。很多数据挖掘过程通常基于相似性或寻找“合适”的相似性来对个体进行分组。本书前面的章节间接地体现了这一点，比如分类模型生成分类边界来将目标变量值相同的个体归为同一组。本章会对相似性进行直接的探讨，同时展示其在不同类型任务上的应用。另外，本章加入了一些介绍技术细节的小节，以便于数学功底较好的读者更深入地理解相似性。不过，读者跳过这些部分也无妨。

许多商业任务中都涉及基于相似个体进行推理的过程。

- 有时我们想直接**找到**相似的个体。比如，IBM 想找到与其最佳商业客户相似的企业作为其销售部门的目标客户。再比如，惠普公司（Hewlett-Packard）维护着许多面向客户的高性能服务器。这种维护往往是由一种工具辅助进行的。而这种工具可以在已知某服务器的配置的情况下，获取其他配置相似的服务器上的信息作为参考。另外，广告商通常希望向与优质老客户相似的新客户提供线上广告。
- 相似性可以用于**分类**和**回归**。鉴于我们现在已经非常了解分类了，所以下文将会用一个分类的示例来展示相似性的用法。
- 有时我们还想把相似的个体归为一簇（即一组），比如我们想知道客户群中是否存在相似客户的类群，以及这些相似客户的共同点是什么。前面章节探讨了有监督的划分，本章讨论的是基于相似性的无监督的划分。在讨论过相似性在分类中的用途后，本章还会讨论它在聚类中的用途。

- 诸如亚马逊和 Netflix 这样的现代零售商利用相似性来推荐相似的商品或基于相似的用户提供**推荐服务**。每当你看到“喜欢 X 的人也喜欢 Y”或“与你浏览历史相同的用户也看了……”这样的商品推荐信息时，相似性都正在被应用。在第 12 章中会看到，当使用相同的“品味维度”来描述时，一名用户和一部电影之间就可能存在相似性。这时候我们可以寻找与某个用户最相似的（并且该用户未观看过的）电影作为针对这个用户的电影推荐。
- 当然，根据相似个体进行推理的过程在其他领域也十分常用。它天然适用于医药和法律等领域。医生可以通过参考相似病例（不论是亲自诊治的还是文献记载的）和诊断结果来对新的棘手病情进行诊断。律师常常援引判例来进行辩护，而这些判例是已经判决并收录进案卷的相似的历史案例。在人工智能领域，辅助医生和律师进行病例 / 案例推理的系统的构建已经有很长一段历史了，而其所依赖的关键因素就是相似性判断。

为便于深入探讨这些应用场景，本章首先要花一点时间来严格地明确一下相似性及一个与其非常相近的概念：距离。

6.1 相似性和距离

只有在对象被表示为数据后，我们才能更精确地讨论对象间的相似性或距离。例如本书中一直在使用的数据表示方式——把每个对象表示为特征向量。在这种方式下，两个对象在由特征定义的空间中距离越近，两者就越相似。

在构建和应用预测模型时，我们的目标是确定目标变量值。为此，我们已经隐性地运用了对象间的相似性：3.3 节中探讨了一些分类模型的几何意义；4.1 节则探讨了两种不同的模型，它们均根据具有相同类别标签的个体的接近程度将实例空间划分成若干区域。数据科学中许多方法都能从这个角度来看：作为组织数据实例（重要对象的代表）空间的方法，为了服务于特定目的，相似的实例会被相似地对待。比如，分类树和线性分类器都能通过构建分区边界来区分不同类别；两种方法都认为同一个分区中的数据点应该是相似的。两者的区别仅在于如何表示和发现分区。

所以，为什么不直接对个体间的相似性或距离进行推断呢？为此我们需要掌握度量相似性或距离的基本方法。比如，当我们说两个企业或两位消费者是相似的时，这究竟意味着什么呢？下面来仔细探讨一下这个问题。首先，考虑两个简化的信贷申请场景中的实例：

属　　性	用户A	用户B
年龄	23	40
当前地址的居住时长（年）	2	10
居住方式（1 = 自有，2 = 租赁，3 = 其他）	2	1

因为这些数据项包含多个属性，所以我们没法把它们归一为某种单一的度量方式。度量用户 A 和用户 B 之间的相似性或距离的方法其实有很多，不妨先从基本的几何开始。

根据前文中讨论过的几何表示方法，由两个（数值）特征描述的任何对象都可以视作一个二维空间中的点。图 6-1 中的两个数据项，A 和 B，就是在这样的二维平面中。其坐标分别为 (x_A, y_A) 和 (x_B, y_B)。虽然前文可能提过数次，但在此仍要再次强调：这些坐标也就是两

个点的特征值。如图所示，我们可以在两点之间画一个直角三角形，其底为两点横坐标的差 $|x_A - x_B|$，其高为两点纵坐标的差 $|y_A - y_B|$。由勾股定理可知，A 和 B 的距离就是该三角形的斜边之长，也即其他两边的长的平方和再开方 $\sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$ 。本质上，我们可以通过计算单个维度（即本例中的单个特征）上的距离来计算出空间中存在的所有距离，这便是两点之间的**欧几里得距离**[1]。这种距离度量方法可能是最常用的几何距离度量方法。

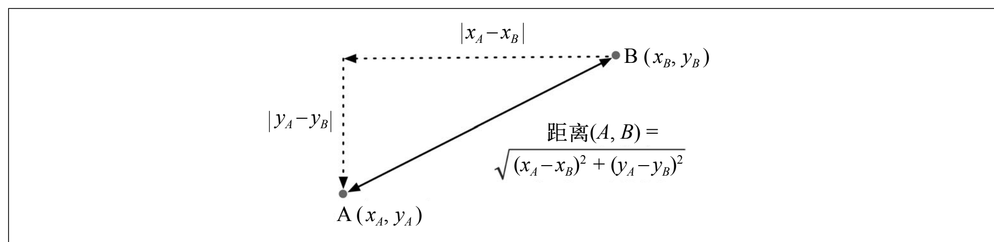


图 6-1：欧几里得距离

欧几里得距离不仅限于二维。A 和 B 如果均包括 3 个特征，则可以表示为三维空间中的点，两者的坐标分别为 (x_A, y_A, z_A) 和 (x_B, y_B, z_B)。那么，A 和 B 的距离项还将包含 $(z_A–z_B)^2$。我们可以加入任意多个特征，每个特征对应一个新的维度。当对象由 n 个特征描述，即处于 n 维空间 $(d_1, d_2, \cdots, d_n)$ 时，该 n 维空间下欧几里得距离的通式如公式 6-1 所示。

公式 6-1：欧几里得距离通式

$$\sqrt{(d_{1,A} - d_{1,B})^2 + (d_{2,A} - d_{2,B})^2 + \cdots + (d_{n,A} - d_{n,B})^2}$$

我们现在有了任意两个（由数值型特征向量描述的）对象间的距离的度量方法——一个基于对象的每一个特征间距离的简单公式。因此前文中的用户 A 和用户 B 之间的欧几里得距离是：

$$\begin{aligned} d(A,B) &= \sqrt{(23-40)^2 + (2-10)^2 + (2-1)^2} \\ &\approx 18.8 \end{aligned}$$

由上式可得，两者的距离约为 19。这是既没有单位也没有具体含义的数字，只能用于比较实例个体两两之间的相似性。事实证明，这种比较十分有意义。

6.2　最近邻推理

既然有了度量距离的方法，就可以用它来解决数据分析工作中的许多问题了。回忆一下本章开头的示例，我们可以用这个方法找到与最佳的企业客户最相似的企业，或与最佳零售客户最相似的线上消费者，找到后，便可以根据商业需要采取相应的措施。IBM 用这种方法指导其销售人员针对企业客户进行营销。线上广告商用这种方法来精准投放广告。这些最相似的实例个体被称为**最近邻**。

注 1：以公元前 4 世纪被称为“几何学之父”的希腊数学家欧几里得的名字命名。

6.2.1 示例：威士忌分析

现在请看一个新的示例。本书作者 Foster 喜欢纯麦苏格兰威士忌。如果你很有经验，就会发现成百上千种的纯麦威士忌各不相同。当 Foster 找到一种他特别喜欢的纯麦威士忌时，他就想再找到其他类似的品种，一是因为他喜欢探索纯麦威士忌的“世界”；二是因为在任何酒类专卖店和餐厅中，这种酒的选择都很有限，而他想找出一种他非常喜欢的。比如，一天晚上，他的饭友推荐他尝尝纯麦“Bunnahabhain”威士忌[2]，这种酒的味道很特别，而且非比寻常地好。那么他如何在所有的纯麦威士忌中找到另一款像“Bunnahabhain”一样的呢？

我们将采取数据科学的方法。第 2 章提到，首先应考虑要回答的问题是什么，以及什么样的数据适合用来回答该问题。如果我们希望能把口味近似的威士忌看作是相似的，那么应该用什么样的特征向量来描述纯麦苏格兰威士忌呢？这正是蒙特利尔大学的 François-Joseph Lapointe 和 Pierre Legendre（1994）所研究的项目。他们对苏格兰威士忌的若干分类问题和组织问题非常感兴趣。此处我们将选用他们的一部分方法。[3]

其实，许多种威士忌都有相关的品尝手记。比如，Michael Jackson 就是一位著名的威士忌和啤酒的鉴赏家。他还撰写过 *Michael Jackson's Malt Whisky Companion: A Connoisseur's Guide to the Malt Whiskies of Scotland*（Jackson, 1989），其中包含了对于 109 种纯麦苏格兰威士忌的描述。这些描述以品尝手记的格式记录，如：“开胃的煤烟香，几乎类似于熏香，带圆润果味的石楠花蜜。”

作为数据科学家，我们已有所进展：已经找到了可能大有用处的数据源。不过由于仅凭这些威士忌的品尝手记，还无法形成用来描述它们的特征向量，因而我们需要对数据形式做进一步转化。受 Lapointe 和 Legendre（1994）的启发，我们根据每种苏格兰威士忌的品尝手记，建立了概括其中信息的数值特征，并据此定义了威士忌的五个通用属性，每个属性都有众多可能取值，如下所示。

(1) 颜色：黄色、非常淡、淡色、淡金、金色、古金色、全金、琥珀色等　（共 14 种取值）

(2) 香味：芳香、泥煤香、甜香、清香、清新、苦香、青草香等　（共 12 种取值）

(3) 口感：柔软、中等、饱满、圆润、柔滑、轻盈、浓厚、油滑　（共 8 种取值）

(4) 滋味：饱满、微苦、雪利酒味、浓烈、果味、青草味、烟熏味、咸味等　（共 15 种取值）

(5) 余韵：饱满、微苦、温和、轻盈、柔滑、甘冽、果味、青草味、烟熏味等(共 19 种取值）

需要指出的是，这些类别值并非互斥（比如，Aberlour 的滋味就可以被描述为中等、饱满、柔软、圆润和柔滑）。通常，这些值可以同时出现（即使其中的几个值永远不可能同时出现，如又淡又浓重的颜色），但正因为它们可以同时出现，所以 Lapointe 和 Legendre 把每个变量的每个值都编码成了单独的特征，因此每种威士忌有 68 个二值型特征。

注 2：他也读不对这个单词。

注 3：基于威士忌分析的真实示例可见 WhiskyClassified.com。

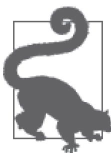
由于 Foster 喜欢喝 Bunnahabhain 威士忌，因而我们可以借助 Lapointe 和 Legendre 的表示法，用欧几里得距离来寻找与之相似的其他威士忌。下文中的 Bunnahabhain 威士忌的描述可供参考。

- 颜色：金色
- 香味：清新、海味
- 口感：浓厚、中等、轻盈
- 滋味：甜味、果味、清冽
- 余味：饱满

下面是对 Bunnahabhain 威士忌，以及其他 5 种与之最为相似的纯麦苏格兰威士忌的描述，按距离从小到大排序。

威士忌种类	距　离	描　述
Bunnahabhain	—	金色；浓厚、中等、轻盈；甜味、果味、甘冽；清新、海味；饱满
Glenglassaugh	0.643	金色；浓厚、轻盈、柔滑；甜味、青草味；清新、青草味
Tullibardine	0.647	金色；浓厚、中等、柔滑；甜味、果味、饱满、青草味、甘冽；甜味；浓厚、芳香、甜味
Ardbeg	0.667	雪利酒香；浓厚、中等、饱满、轻盈；甜香；苦味、泥煤味、海味；咸味
Bruichladdich	0.667	淡色；浓厚、轻盈、柔滑；苦味、甜味、烟熏味、甘冽；轻盈；饱满
Glenmorangie	0.667	淡金；中等、油滑、轻盈；甜香、青草香、辛辣；甜味、辛辣、青草味、海味、清新；饱满、悠长

我们可以用这个表格找出与 Bunnahabhain 相似的苏格兰威士忌。购买威士忌时，虽然我们可能需要在店内库存中寻找表格中的项，但是由于表格是按相似性排序的，因而我们可以轻而易举地找到库存中与 Bunnahabhain 最为相似的威士忌（还能通过与没有库存的其他选项比较，大体了解该种威士忌与 Bunnahabhain 有多相似）。



如果你对苏格兰威士忌数据集很感兴趣，那么不妨访问 http://adn.biol.umontreal.ca/~numericalecology/data/scotch.html，查看 Lapointe 和 Legendre 的数据以及论文。

这个例子直接应用了相似性来解决问题，一旦理解了这一基本概念，它就能作为一个强大的概念性工具来解决许多问题，比如上文所展示的那些问题（寻找相似企业、相似消费者等）。威士忌示例告诉我们，为了保证相似性与有意义的特征挂钩，数据科学家往往需要进一步定义数据。本章后文将阐述其他有关相似性和距离的概念，而现在，先讨论相似性在数据科学中的另一种常见用法。

6.2.2　用最近邻来进行预测建模

我们还可以用最近邻的概念来进行预测建模。回忆一下你在前几章中学到的所有有关预测建模的知识。在预测建模中运用相似性的基本过程非常简单：给定一个目标变量未知的新实例，在浏览所有训练实例后，选择其中与新实例最为相似的那些，然后根据这些实例的

目标变量值（已知）来预测新个体的目标变量值。执行最后一步的方法仍需进一步明确，不过目前我们先称之为运用在近邻的已知目标变量值上的、能帮助进行预测的**合成函数**（如投票或取平均值）。

1. 分类

由于目前为止本书花了很大篇幅来解决分类问题，因而本节先讨论最近邻方法在极简单的场景下对新实例进行分类的方法。图 6-2 中标记为“？”的新实例的标签需要预测，根据上文介绍的基本过程，我们找到了它的最近邻（本例中有三个）及它们的已知的目标变量值（类别值）——两个为正，一个为负。那么应该怎么构造组合函数呢？适合本例的一个简单函数是多数票决，根据这种方法，该实例的类别应该也为正。

再考虑一个稍微复杂些的信用卡营销问题。我们的目标是，基于相似客户对信用卡优惠活动的响应情况，预测新客户的响应情况。其数据（当然，也是极度简化了的）展示在表 6-1 中。

表6-1：最近邻示例：David会不会响应？

客户	年龄	收入（万）	信用卡数量	是否响应（目标变量）	与David的距离
David	37	5	2	?	0
John	35	3.5	3	是	$\sqrt{(35-37)^2+(35-50)^2+(3-2)^2}=15.16$
Rachael	22	5	2	否	$\sqrt{(22-37)^2+(50-50)^2+(2-2)^2}=15$
Ruth	63	20	1	否	$\sqrt{(63-37)^2+(200-50)^2+(1-2)^2}=152.23$
Jefferson	59	17	1	否	$\sqrt{(59-37)^2+(170-50)^2+(1-2)^2}=122$
Norah	25	4	4	是	$\sqrt{(25-37)^2+(40-50)^2+(4-2)^2}=15.74$

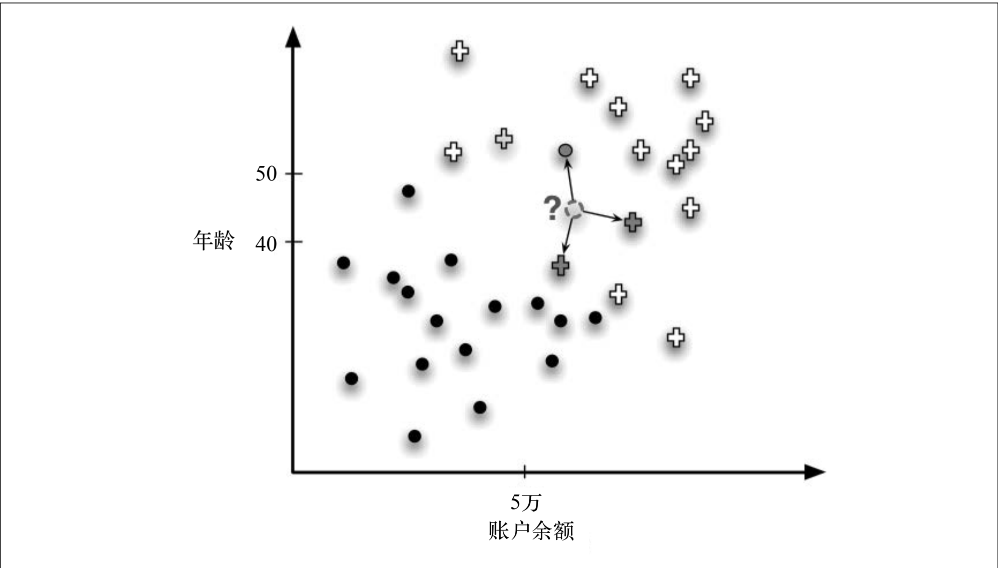


图 6-2：最近邻分类。“？”是需要分类的点，其类别应为“+”，因为与之最近的（三个）近邻大多为“+”

示例数据中的 5 个用户曾收到过该信用卡优惠活动邀请，而我们所掌握的有关他们的信息有姓名、年龄、收入、已开通的信用卡数和是否对活动做出了响应。现在我们想预测新客户 David 是否会对活动做出响应。

表 6-1 的最后一列是用公式 6-1 计算出的距离值，表示每个客户与 David 的相似程度。其中 3 个客户（John、Rachael 和 Norah）与 David 非常相似，距离约为 15，而其他 2 个客户（Ruth 和 Jefferson）则距离更远。因此，David 的 3 个最近邻按距离由近到远排序，分别是 Rachael、John 和 Norah。Rachael 做出了响应，John 和 Norah 则没有。如果采取多数票决方式，那么预测结果就为“是”（David 会做出响应）。这引出了最近邻方法的一些重要问题：应该选择多少个最近邻？它们在组合函数中的权重相等吗？我们将在本章后文中讨论这些问题。

2. 概率估计

前文提过，不仅预测新个体的分类很重要，预测其类别的概率——即确定其分数——也很重要。这是因为分数所包含的信息比单纯的“是 / 否”二元结果更丰富。最近邻分类可以轻而易举地达到这个要求。还是刚刚预测 David 是否会响应的分类问题，David 的最近邻（Rachael、John 和 Norah）的目标变量值分别为“否”“是”和“是”，如果给“是”赋值为 1、给“否”赋值为 0，那么三者的平均值 **2/3** 便可以作为 David 的目标变量预测值。实际情况中，我们可能会用更多的最近邻来计算该概率估计值（可回看 3.5 节中有关小样本概率估计的探讨）。

3. 回归

在找到最近邻后，就可以将它们用不同方式进行组合来解决预测问题。刚刚我们用多数票决方法判定了目标变量的值，解决了分类问题，而回归也能同样处理。

假设我们有和表 6-1 一样的数据集，但这次我们想预测 David 的收入。无须重新计算距离，假设 David 的 3 个最近邻仍是 Rachael、John 和 Norah，他们的收入分别为 50 000、35 000 和 40 000。这些值取平均数（约为 42 000）或中位数（40 000）后，可以用来预测 David 的收入。



不得不提的是，我们不会用最近邻的目标变量值来计算距离，因为这是所要预测的项。故此处的收入不会像在表 6-1 中一样被用来计算距离。但我们可以用其他任何已知值来计算距离。

6.2.3 近邻的数量及其影响

在解释分类、回归和评分的过程时，我们仅用了示例中的 3 个最近邻。你可能会提出几个问题。首先，为什么选了 **3** 个最近邻，而非 1 个、5 个或 100 个？其次，这些最近邻的重要性都相同吗？尽管它们都叫“最”近邻，然而有的比其他的更近些，这对它们的重要性有影响吗？

计算所使用的最近邻的数目没有单一标准，但奇数更能避免二元类问题中多数票决方式的平局问题。最近邻算法通常简写为 *k*- 最近邻，其中 *k* 指代所选取的最近邻数，如 3- 最近邻。

一般情况下，k 越大，得到的平均估计值越平滑。如果你理解了目前为止的所有内容，那么你应该也可以理解，如果 k 取最大值（即 $k = n$），那么每次预测都会用到整个数据库。问题在于，这会对整个数据集的目标变量值取平均数作为新个体的目标变量值。该种方法下，分类问题中新个体的目标变量预测值会是整个数据集中占多数的类；回归问题中新个体的目标变量值是所有目标变量值的平均值；类概率估计问题中新个体的目标变量值则是“基础比率”概率值（参考 5.3.1 节中的基础比率注释）。

即使确定了最近邻的个数，但这些最近邻与我们想要预测的个体的相似程度不尽相同，这是否会对它们的重要程度造成影响？

我们起初简单地用**多数**票决方法处理分类问题。为防止出现平局情况，最近邻的个数取奇数。但该方法忽略了很重要的一点：每个最近邻与该个体的距离有多远。比如，假如我们选取了 David 的 4 个最近邻，其响应情况分别为“是”“否”“是”“否”，造成了平局。但前三个距离 David 非常近（距离约为 15），而相比之下，第四个非常远（距离约为 122），直观地看，第四个客户在票决中所占比重不应与前三个一样大。考虑到这个问题，最近邻方法往往采取**加权表决**或**相似性适度投票**，从而让每个近邻的贡献度与相似程度挂钩。

请再次思考表 6-1 中的数据，并判断 David 是否会对信用卡优惠活动做出响应。前面已经证明，如果用多数票决方式来预测 David 的类别，则需特别注意最近邻的个数。因此我们这次把所有最近邻按与 David 的相似程度加权后重新计算，其中权重为距离平方的倒数。下面是按距离对最近邻进行排序的表格。

姓　　名	距　　离	相似性权重	贡献度	类
Rachael	15.0	0.004 444	0.344	否
John	15.2	0.004 348	0.336	是
Norah	15.7	0.004 032	0.312	是
Jefferson	122.0	0.000 067	0.005	否
Ruth	152.2	0.000 043	0.003	否

“贡献度”一栏指的是每个最近邻对 David 最终的目标变量概率预测的贡献量（与权重成比例，和为 1）。从中可以看出，贡献度在很大程度上受距离影响：Rachael、John 和 Norah 与 David 最为相似，因而是 David 响应情况的预测值的主力；而 Jefferson 和 Ruth 相对远些，几乎没有对 David 的预测做出贡献。把两类的贡献值分别相加，最终 David 的概率预测是 0.65 的“是”和 0.35 的“否”。

这个概念也可以推广到其他类型的预测方法上，如回归和类概率估计。通常，我们可以把该过程视作加权**评分**。加权评分有一个好处，就是削弱了决定最近邻个数的重要性。因为每个最近邻的贡献度与距离挂钩，所以越远的近邻自然影响越小，故而在使用加权评分法时，k 的取值不像在多数票决法或未加权平均法中那样关键。许多方法就是通过选择一个很大的 k（比如 $k = n$，即选择所有数据点），且用距离调整其影响来避免过多考虑 k 的取值。

> **最近邻推理的多种名称**
>
> 像数据挖掘领域的很多名词一样，最近邻分类器也有许多不同叫法，部分原因是许多独立的领域都产生了相近的概念。最近邻分类器在很久以前诞生于统计和模式识别（Cover & Hart, 1967）领域，通过参考数据库（或称“记忆”）中的数据对新个体直接进行分类的概念当时被称作**基于实例的学习**（Aha, Kibler & Albert, 1991）或**基于记忆的学习**（Lin & Vitter, 1994）。由于最近邻分类器在“训练”阶段不产生模型，而是直到获取所需个体后才进行主要的工作，因而这种总体思路也叫**惰性学习**（Aha, 1997）。
>
> 人工智能领域中的一项相关技术是**案例推理**（Kolodner, 1993; Aamodt & Plaza, 1994），简称 CBR。因为医生和律师往往会根据过往案例来推断新案例，所以该技术在这些领域已经具有相当长的历史了。
>
> 然而，案例推理和最近邻方法之间仍存在显著差异。CBR 中的案例通常不是简单的特征变量的形式，而是对该案例非常详尽的综述，包含诸如症状、病史、诊断、治疗和结果等内容；或是法律案例的细节，如原告论据和被告论据、引用的先例和最终判决结果等。这些案例纤悉必具，因此在 CBR 中，它们不仅能用来预测类标签，还能提供诊断和规划信息以便后用。在新情景下应用过往案例这一过程往往十分复杂，需要付出巨大努力。

6.2.4 几何解释、过拟合和复杂度控制

像我们学过的其他模型一样，对最近邻方法创建的分类区域进行可视化是很有意义的。虽然没有明确的边界，但是由实例间的相邻关系构造的隐性区域确实存在。通过系统地探索实例空间中的数据点、判定其分类以及在分类变化的位置设定边界，我们便可以计算出这些区域。

图 6-3 描绘了一个由 1- 最近邻分类器构造的、由“无不良贷款”个体组成的区域。试将此图与图 3-15 中的分类树区域以及图 4-3 中线性边界包围的区域作比较。

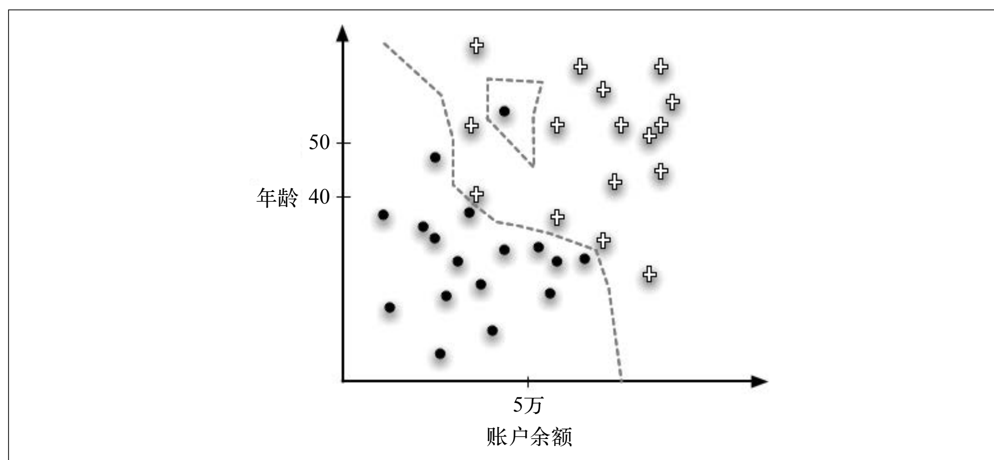


图 6-3：由 1- 最近邻分类器构造的边界

注意，这里的边界不是线条，也不是简单的几何图形。它们是不同类别的训练数据之间的不规则边界。最近邻分类器沿着训练集实例周边构造了具体边界。注意，一个在正个体群中间的负个体产生了一个“负岛”，该点可以被当作噪声或离群点。而使用其他建模方法的话，该点可能会被消除。

这种对异常值的敏感性源于我们使用了 1- 最近邻分类器。由于其只选择单个近邻，因而所产生的边界要比由多个近邻取平均产生的边界更不规则一些。我们稍后会继续这个话题。一般来说，不规则概念边界是所有最近邻分类器的特征，因为它不受任何特定的几何形式限制，而是完全根据训练数据来构造边界。

由此我们可以想到第 5 章中有关过拟合和复杂度控制的讨论，如果你猜想 1- 最近邻分类器具有十分严重的过拟合问题，那你就对了。可以设想一下基于训练集评估 1- 最近邻分类器的结果——在对每个训练数据点分类时，使用任何合理的距离度量方法都会导致该点成为自己的最近邻！那么该点的目标变量值就会用来预测自身的目标变量值，然后，你瞧，完美的分类就出现了。回归方法也会有同样的问题。1- 最近邻分类器会记住训练数据，但效果会比第 5 章开头那个站不住脚的的查询表稍好一点，因为查询表不含任何相似性的概念，所以它只会完美预测某个特定的训练个体，而对其他个体则给予一样的默认预测。1- 最近邻分类器同样能完美地预测训练个体，但经常也可以对其他个体做出合理预测：因为它用的是与之最相似的训练个体。

因此，从过拟合及避免过拟合的角度来看，*k*- 最近邻分类器中的 k 是一个复杂度参数。在 $k = n$ 的极端情况下，模型的复杂度大大受限。根据前文描述，*n*- 最近邻模型（忽略相似性权重）仅能根据数据集中的目标变量平均值对每个数据点进行预测。在另一个极端，即 $k = 1$ 的情况下，我们会得到一个非常复杂的模型，而其构造的边界也非常复杂，这样会使每个训练个体处在一个由它自身的类别标注的区域里。

现在回到先前的一个问题：如何确定 k 的取值？我们可以用 5.9.2 节中的程序来设置其他复杂度参数：先在训练集上做交叉验证或其他嵌套保留测试，从而在大量不同的 k 的取值中选出使模型效果最佳的值，再用整个训练集来构建 *k*- 最近邻模型。第 5 章中详细讲述过，因为该程序只用了训练数据，所以我们便可以用测试数据来对其进行评估，从而得到对其泛化能力的无偏估计。数据挖掘工具通常具有这种用嵌套交叉验证来自动确定 k 值的功能。

图 6-4 和图 6-5 展示了最近邻分类器创建的不同边界。一个三元分类问题使用不同的最近邻数进行了分类：图 6-4 中的最近邻数为 1，于是图中的边界非常不规则，且具体到了训练集中的每个数据点；而图 6-5 中的最近邻数为 30，它们取平均数后得到了最终的分类结果，因此图中的边界与图 6-4 中的边界截然不同且更为平滑。需要注意的是，这两个案例皆不同于线性模型或树形结构模型，它们既不产生平滑的曲线形边界，也不产生规则的几何形分区。*k*- 最近邻模型的边界与数据的关联性更高。

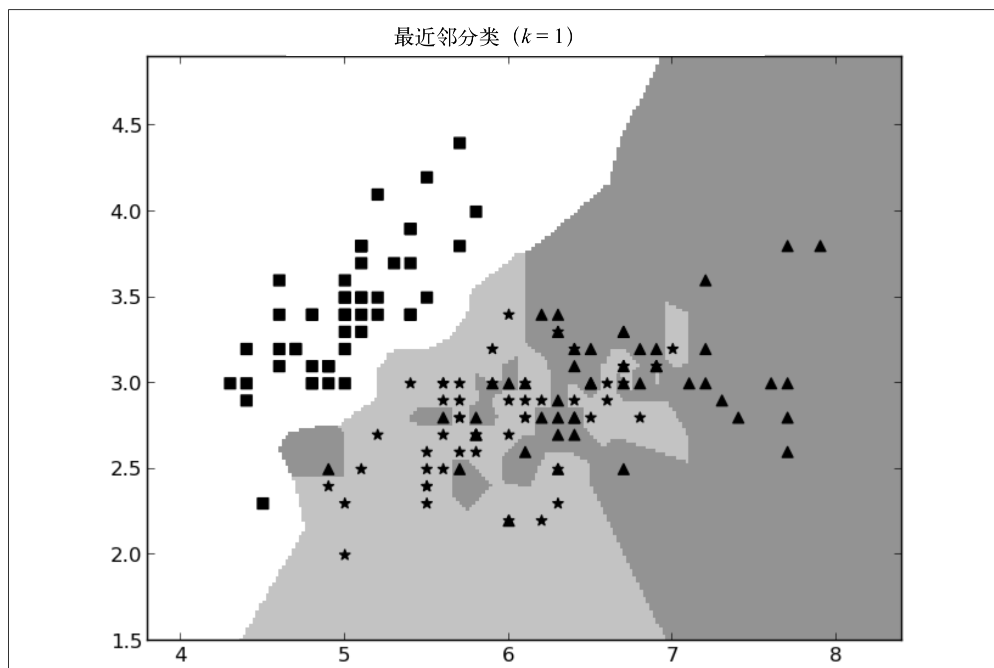


图 6-4：针对三元分类问题，用 1- 最近邻分类器（1 个最近邻）构建的分类边界

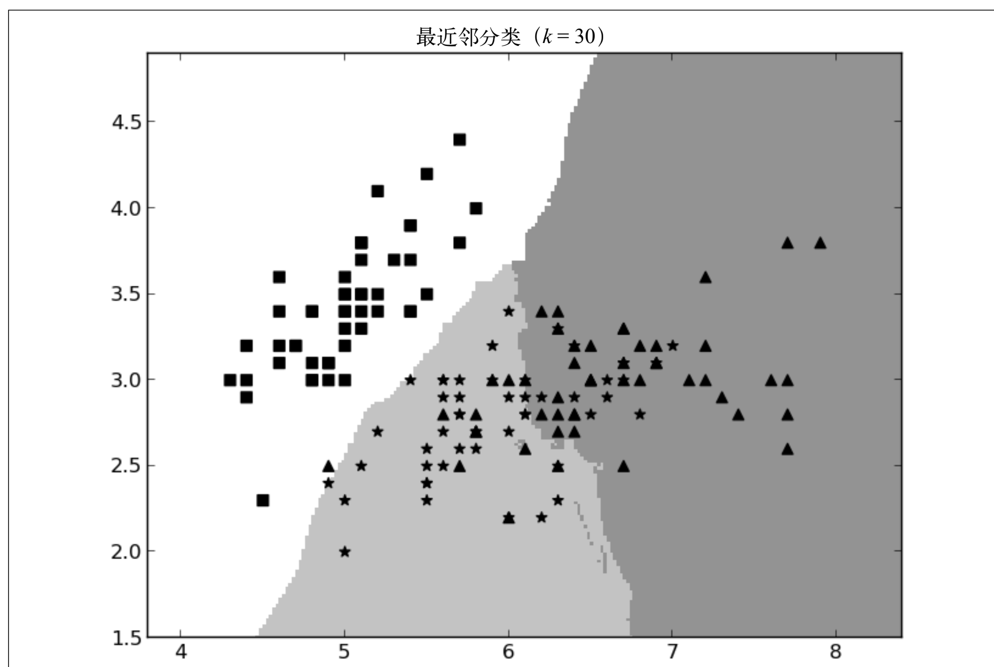


图 6-5：针对三元分类问题，用 30- 最近邻分类器（对 30 个最近邻取平均）构建的分类边界

6.2.5 最近邻方法的问题

在结束对作为预测模型的最近邻方法的讨论之前，我们需要提出有关该模型的使用方法的几个问题。这些问题在现实应用中经常出现。

1. 易理解性

最近邻分离器的易理解性是个很复杂的问题。前文提过，诸如医药和法律之类的领域通常会通过相似的过往案例来推断新案例，在这些领域里，最近邻方法是个不错的选择。但在其他领域，这种不够明确且不太容易解释的模型可能会造成问题。

易理解性的问题实际上又分为两个方面：做出特定**判断**的理由和整个**模型**的易理解性。

在使用 *k*- 最近邻方法时，我们可以轻松地描述出单个数据点的目标变量是如何被决定的：我们可以展示出决策所用到的最近邻集合，以及每一个最近邻的贡献度，如前文表 6-1 中预测 David 是否响应的示例。使用最近邻方法时，相应的谨慎解释和明确描述非常有用，如 Netflix 使用最近邻分类来进行推荐，并使用类似下面的语句来解释为何推荐这些电影：

> “根据您喜欢的《莫扎特传》《不朽的园丁》和《阳光小美女》，我们为您推荐《舞出我天地》。”

亚马逊的推荐语则像这样：“与您的搜索记录相近的用户购买了……”“……与您浏览过的商品相关”。

这样的推荐理由是否充分要依应用场景而定。一个亚马逊用户可能会对这样的推荐理由感到非常满意。但在另一种情形下，一位抵押贷款申请人则可能对诸如“因为你与曾经贷款违约的 Smith 一家和 Mitchell 一家非常相似，所以我们拒绝你的申请”的解释感到不满。这种模型可以基于特定的重要变量而给出极简单的解释。实际上，这种模型如果被用于信用评分模型的话，会受到某些法规的限制。比如，某个线性模型可以被这样解释：“其他条件不变的情况下，如果你的收入高于 20 000 美元，那么你就能得到这笔贷款。”

同样，整个最近邻模型判定新个体的方法也非常容易解释，这种通过寻找最相似的实例并以它们的分类或值来进行预测的思路，对许多人来说都非常直观。

难点在于，如何更深入地解释从数据中挖掘出的“知识”。如果企业利益相关者提出“你的系统从数据中获得了有关我用户的什么信息？做出判断的依据是什么？”这样的问题，那你可能无法轻松地给出答案，因为该模型不是显式的。严格来讲，最近邻“模型”包含了整个案例集（数据库）、距离函数和组合函数。在二维空间中，我们可以直接对该模型进行可视化（如前一幅图所示）；但维度更高时，这种可视化就无法进行。因为该模型中的知识通常不易理解，所以当模型的易理解性和依据非常重要时，不建议使用最近邻方法。

2. 维度和领域知识

最近邻方法通常会用所有的特征来计算实例之间的距离。6.3.1 节探讨了有关属性的一个难题：数值属性的值域可能存在巨大差异，因而如果没有经过合理的标准化，那么值域较大的变量可能会覆盖值域较小的变量的效果。除此之外，当属性过多，或与判断相似性不相关的属性过多时，也会存在严重的问题。

比如，在信用卡优惠活动问题中，用户数据库中可能包含许多附带信息，如子女数、工作时长、房屋大小、收入中位数、汽车的品牌及型号、平均教育水平等，这些变量有的或许与用户是否会接受优惠活动相关，但大多数无关。这样的问题被称为高维问题，即它们会受到所谓**维度灾难**的影响，而这也给最近邻方法带来了很多问题。其原因和影响具有一定技术性[4]，但粗略地说，因为所有属性（维度）都被用来计算距离，所以实例的相似性会大大地被过多的无关变量所误导或扰乱。

解决无关属性过多的问题的方法有很多，其中一个是**特征选择**，即审慎地选择应进入数据挖掘模型的特征。数据挖掘人员可以借助背景知识，手动选择有关系的属性。这是数据挖掘团队在数据挖掘流程中注入大量领域知识的主要方式之一。第 3 章和第 5 章探讨过，一些自动的特征选择方法也能处理数据，并判断哪些属性给出了有关目标变量的信息。

另一种在相似性计算中注入领域知识的方法是手动调整相似性 / 距离函数。比如，数据科学家可能预先知道"**信用卡数量**"对用户是否会再办一张新卡有很大的影响，那么他就可以通过赋给不同的特征不同的权重（比如，赋给**信用卡数量**更大的权重）来调整距离函数。加入领域知识不仅是因为我们知道如何提升预测效果，更是因为我们了解正在寻找的相似个体。在寻找相似的威士忌时，我可能事先知道"泥煤味"对我找口味相似的纯麦威士忌非常重要，那么我就可以在计算相似性时给其以更高的权重，而如果另一个口味变量不太重要，那么我就可以删掉它，或是给它一个较低的权重。

3. 计算效率

最近邻方法的优势之一是训练速度快，因为其仅需要对个体进行存储，而无须构建模型。其主要的计算成本在于计算和分类，因为我们必须通过查询数据库来找寻新个体的最近邻。这个过程可能代价不菲，且分类阶段的成本也需要多加考虑。有的应用场景要求极快的预测速度，比如，线上广告精准投放就要求在几十毫秒内做出决策。在这种情况下，最近邻方法就不够实用了。



提取最近邻的速度可以用一些技术来提高。一些商业数据库和数据挖掘系统会用 kd 树和散列方法（Shakhnarovich, Darrell & Indyk, 2005; Papadopoulos & Manolopoulos, 2005）等专门的数据结构提升最近邻查询的效率。但请注意，许多小型研究型数据挖掘工具通常不会使用这样的技术，而是仍依靠简单的暴力检索方法。

6.3　与相似性和最近邻相关的一些重要技术细节

6.3.1　混合属性

到目前为止，我们一直在使用欧几里得距离，并证明了其计算的简便性。如果变量是数值型的且可以直接比较，那么距离的计算就很简单。而当数据点包含复杂且混合的变量时，问题也会变得复杂。思考一下同一问题的另一个示例，这其中包含更多的属性：

注 4：比如，出于技术性原因，在特征很多的时候，一些特征实例会极其频繁地出现在其他实例的 k 个最近邻中，因此这些个体会对分类造成巨大影响。

属　　性	客户A	客户B
性别	男	女
年龄	23	40
当前地址的居住时长（年）	2	10
居住方式（1 = 自有，2 = 租赁，3 = 其他）	2	1
收入（美元）	50 000	90 000

现在我们面临一些问题：首先，欧几里得距离公式的项均为数值型，但“性别”是类别型（或符号别型）变量，因此必须用数字对其进行编码。像这样的二元变量往往用 0-1 编码，但这对多元类别型变量来说并非最佳的编码方式。

另一个重要问题是，有些变量虽然是数值型，但其取值范围大不相同。年龄的范围是 18 到 100 岁，而收入的范围则可能是 10 到 1000 万美元。在标准化前，10 美元的收入差可能和 10 岁的年龄差在距离尺度中同等重要，这显然是错误的。因此，基于最近邻方法的系统通常首先要经过变量的标准化：通过测量变量的范围来对值进行相应的标准化，或把值分配到固定数目的分箱中。这里的一般原则是，在计算相似性或距离时，必须要谨慎处理，要注意计算方式对所应用的问题的真正意义。

6.3.2　*其他距离函数



前方有技术细节！

简单起见，到目前为止我们只用了一种度量方法——欧几里得距离。下文将介绍距离函数的更多细节，以及其他度量方法。

值得注意的是，这里提到的相似性测度只是冰山一角。虽然它们尤为常用，但数据科学家和商业分析师必须谨记，应该根据实际商业问题来选择有意义的相似性测度。跳过本节不会影响后面的阅读。

前文提过，欧几里得距离（Euclidean distance）可能是数据科学领域应用最广的距离度量方法。它通用、直观并且计算起来很快。由于它在每个维度使用距离的**平方**，因而有时也叫“L2 范数”，记作“$\|\cdot\|_2$”。公式 6-2 展示了它的标准公式。

公式 6-2：欧几里得距离（L2 范数）

$$d_{\text{Euclidean}}(\boldsymbol{X},\boldsymbol{Y}) = \|\boldsymbol{X}-\boldsymbol{Y}\|_2 = \sqrt{(x_1-y_1)^2+(x_2-y_2)^2+\cdots}$$

在欧几里得距离得到广泛应用的同时，仍有许多其他的距离度量方法不容忽视。由 Deza & Deza 编纂的 *Dictionary of Distances*（Elsevier Science, 2006）中列举了几百种距离度量方法，其中有十几种在数据挖掘中非常常用。之所以有这么多种，是因为距离函数在最近邻方法中举足轻重。它基本上可以把两个（可能非常复杂的）示例之间的比较简化为一个数字。应用中的数据类型和领域特征会在很大程度上影响单个属性组合方式之间的差异。

曼哈顿距离（**Manhattan distance**）（或称 **L1 范数**）是两个数据点不同维度上的距离（**非平方项**）的和，如公式 6-3 所示。

公式 6-3：曼哈顿距离（L1 范数）

$$d_{\text{Manhattan}}(\boldsymbol{X},\boldsymbol{Y}) = \|\boldsymbol{X}-\boldsymbol{Y}\|_1 = |x_1 - y_1| + |x_2 - y_2| + \cdots$$

该公式将 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 在每个维度上的差值简单相加。它之所以被称为曼哈顿距离（或出租车距离），是因为它表示了一个人在类似曼哈顿区中心（网格状形式）这样的地方的两点之间移动时，他所走过的所有街道的总距离，即总的横向距离加上总的纵向距离。

上文中威士忌问题的研究者用的是另一种常用的距离度量方法[5]——**杰卡德距离**（**Jaccard distance**）。它能把两个对象作为特征集合进行处理。在这种思维方式下，我们需要考虑两个对象 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的所有特征的并集 $|\boldsymbol{X} \cup \boldsymbol{Y}|$ 和交集 $|\boldsymbol{X} \cap \boldsymbol{Y}|$，它们的杰卡德距离是两者共有的特征数与两者全部的特征（两者中任意一个所拥有的特征）数之比。当两者共有的特征更重要，而两者同时**缺少**的特征不重要的时候，杰卡德距离比较适用。比如，对寻找相似的威士忌来说，两种威士忌都有泥煤味很重要，但两者都没有咸味则不重要。杰卡德距离的集合表示法如公式 6-4 所示。

公式 6-4：杰卡德距离

$$d_{\text{Jaccard}}(\boldsymbol{X},\boldsymbol{Y}) = 1 - \frac{|\boldsymbol{X} \cap \boldsymbol{Y}|}{|\boldsymbol{X} \cup \boldsymbol{Y}|}$$

余弦距离（**cosine distance**）则常在文本分类中用于度量两篇文档的相似性，其定义见公式 6-5。

公式 6-5：余弦距离

$$d_{\text{cosine}}(\boldsymbol{X},\boldsymbol{Y}) = 1 - \frac{\boldsymbol{X} \cdot \boldsymbol{Y}}{\|\boldsymbol{X}\|_2 \cdot \|\boldsymbol{Y}\|_2}$$

其中 $\|\cdot\|_2$ 仍代表每个特征向量的 L2 范数，或称欧几里得长度（对向量而言，这仅为其到原点的距离）。



信息检索的文献中更常使用**余弦相似性**这一说法，即公式 6-5 的分数部分，也即 1- 余弦距离。

在文本分类中，每个词和记号都对应一个维度，文章在每个维度上的位置则指的是文章中每个词出现的次数。比如，假设**表演**一词在文章 A 中出现了 7 次，在文章 B 中出现了 2 次；**过渡**一词在 A 中出现了 3 次，在 B 中出现了 3 次；**金融**一词在 A 中出现了 2 次，在 B 中则没有出现。如果把两篇文章表示成这三个词出现次数的向量形式，即 $\boldsymbol{A} = \langle 7,3,2 \rangle$，$\boldsymbol{B} = \langle 2,3,0 \rangle$。那么两篇文章的余弦距离就是：

注 5：Lapointe 和 Legendre（1994），第 3 节“Classification of Pure Malt Scotch Whiskies”更详细地讨论了他们界定问题的过程，读者可在线上（http://www.dcs.ed.ac.uk/home/jhb/whisky/lapointe/text.html）阅读。

$$
\begin{aligned}
d_{\text{cosine}}(\boldsymbol{A},\boldsymbol{B}) &= 1-\frac{<7,3,2>\cdot<2,3,0>}{\left\|<7,3,2>\right\|_2\cdot\left\|<2,3,0>\right\|_2}\\
&=1-\frac{7\cdot 2+3\cdot 3+2\cdot 0}{\sqrt{49+9+4}\cdot\sqrt{4+9}}\\
&=1-\frac{23}{28.4}\approx 0.19
\end{aligned}
$$

余弦距离尤其适用于需要忽略实例间尺度差异的情况——技术上说，也就是需要忽略向量的幅度的情况。举一个具体的例子，在进行文本分类时，你只想比较两篇文章的内容，而忽略文章长度不同的问题。在上文的例子中，假设还有第三篇文章 C，其中“**表演**”一词出现了 70 次，“**过渡**”一词出现了 30 次，“**金融**”一词出现了 20 次，那么 C 对应的向量就是 $\boldsymbol{C}$ = <70,30,20>。只要稍加计算，你就会发现，$\boldsymbol{A}$ 和 $\boldsymbol{C}$ 的余弦距离是 0，因为 $\boldsymbol{C}$ 恰恰是 $\boldsymbol{A}$ 的 10 倍。

这是阐述距离度量方法多样性的最后一个例子。请试着用另一种方式考虑文本。有时候我们会需要度量两个字符串之间的距离，比如在某些商业应用场景中判断两条数据记录是否对应同一个人，当然，其中可能包括拼写错误。我们最终想知道两者的相似程度。假设有两个字符串：

(1) `1113 Bleaker St.`

(2) `113 Bleecker St.`

我们想知道两者的相似程度，因此需要用另一种距离函数——**编辑距离**或**莱文斯坦距离**。这种度量方法通过计算将一个字符串转化为另一个字符串需要进行的编辑（插入、删除、替换字符中的任意一种）次数的最小值来度量个体间的距离。在本例中，第一个字符串可以通过以下步骤转化为第二个字符串：

(1) 删除一个“1”；

(2) 插入一个“c”；

(3) 把一个“a”换成“e”。

这样这两个字符串的编辑距离就为 3。在其他领域中也可以做相似的编辑距离计算，如姓名（从而可以处理中间名缩写缺失的情况），甚至还可以计算组合了多种编辑距离相似性的更高级别的相似性。



编辑距离也常用于生物领域，以计算等位基因串的遗传距离。一般来说，如果数据项包含需要在意顺序的字符串或序列，那么我们通常会使用编辑距离。

6.3.3 *组合函数：计算近邻的评分



前方有技术细节！

为了完整，我们需要简要探讨一下“组合函数”，即可以通过个体的一系列最近邻计算该个体预测值的公式。

先从简单的多数票决讲起。该决策法则如公式 6-6 所示。

公式 6-6：多数票决分类

$$c(\boldsymbol{x}) = \underset{c \in \text{classes}}{\arg\max}\ \text{score}(c, \text{neighbors}_k(\boldsymbol{x}))$$

其中“class”表示“类”，“score”表示“评分”，下同。此处 $\text{neighbors}_k(\boldsymbol{x})$ 返回的是个体 $\boldsymbol{x}$ 的 k 个最近邻，arg max 返回的是使下一个量达到最大值的参数（此例中指的是 c）。得分函数的定义如公式 6-7 所示。

公式 6-7：多数票决得分函数

$$\text{score}(c, N) = \sum_{\boldsymbol{y} \in N} [\text{class}(\boldsymbol{y}) = c]$$

此处如果 $\text{class}(\boldsymbol{y}) = c$，那么 $[\text{class}(\boldsymbol{y}) = c]$ 的值就为 1，否则为 0。

6.2.3 节中探讨的相似性适度投票，就可以用加入权重的公式 6-6，也即公式 6-8 来完成。

公式 6-8：相似性适度分类

$$\text{score}(c, N) = \sum_{\boldsymbol{y} \in N} w(\boldsymbol{x}, \boldsymbol{y}) \times [\text{class}(\boldsymbol{y}) = c]$$

其中 $\boldsymbol{w}$ 是基于 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 的相似性的权重函数。距离平方的倒数非常常用：

$$w(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{\text{dist}^2(\boldsymbol{x}, \boldsymbol{y})}$$

其中 dist 是该领域中使用的任何一种距离函数。

将公式 6-6 和公式 6-8 转化后，我们可以轻松地输出用于进行概率估计的评分。而由于后者已经输出了评分，因而我们仅需将该评分按所有近邻（neighbors）贡献的总分数标准化，使之介于 0 和 1 之间，如公式 6-9 所示。

公式 6-9：相似性适度评分

$$p(c \mid \boldsymbol{x}) = \frac{\sum_{\boldsymbol{y} \in \text{neighbors}(\boldsymbol{x})} w(\boldsymbol{x}, \boldsymbol{y}) \times [\text{class}(\boldsymbol{y}) = c]}{\sum_{\boldsymbol{y} \in \text{neighbors}(\boldsymbol{x})} w(\boldsymbol{x}, \boldsymbol{y})}$$

最后，只需再做一步，该公式就可以推广到回归中了。回忆一下，回归问题不是在估计新实例 $\boldsymbol{x}$ 的类别，而是通过在函数 f 中输入 $\boldsymbol{x}$ 近邻的一些值来估计 $f(\boldsymbol{x})$ 的值。我们只需把公式 6-9 里括号内包含类的部分换成数值，就能计算出近邻目标变量的加权平均值（组合函数因应用场景而异，有可能需要替换成中位数等），也就是回归估计值。

公式 6-10：相似性适度回归

$$f(\boldsymbol{x}) = \frac{\sum_{\boldsymbol{y} \in \text{neighbors}(\boldsymbol{x})} w(\boldsymbol{x}, \boldsymbol{y}) \times t(\boldsymbol{y})}{\sum_{\boldsymbol{y} \in \text{neighbors}(\boldsymbol{x})} w(\boldsymbol{x}, \boldsymbol{y})}$$

其中 $t(\boldsymbol{y})$ 是实例 $\boldsymbol{y}$ 的目标变量值。

所以，假如要根据潜在用户的某些特征来估计其预期支出，我们可以用公式 6-10，通过计算其近邻的历史支出额的距离加权平均值得出结果。

6.4　聚类

本章开头提出，相似性和距离的概念是数据科学领域中许多内容的基础。为了加深对这些概念的理解，我们来探讨另一种截然不同的任务。回忆一下我们深入学习的第一个数据科学应用：有监督的划分——根据某些我们关心的目标变量的不同取值对个体进行分组。比如，根据合约到期后离开公司的倾向不同对用户进行分组。这里有一个问题：为什么在谈论有监督的划分时，总是使用修饰词"有监督"呢？

有时候我们则会在没有预设的目标特性的情况下给个体——比如用户——分组。他们是否天然地归属于不同分组？需要弄清楚这一点的原因有很多。比如，有时候我们需要从更广阔的视角来回顾一下营销活动的能效——我们是否了解用户？能否在了解用户自然存在的分组后，开发出更好的产品、开展更好的营销活动、采用更好的销售手段和提供更好的客户服务？从数据中发现天然分组的概念叫作无监督的划分，或简称为**聚类**。

聚类是相似性这一基本概念的另一种应用。其基本思路是，找出个体（如用户、企业、威士忌等）的某种分组，使得同一组内的个体之间相似，不同组内的个体之间不相似。



有监督建模方法是基于目标变量值已知的数据来发现能够预测特定目标变量的值的模式。无监督建模则不关注目标变量，而是寻找数据中其他形式的规律。

6.4.1　示例：威士忌分析回顾

在详细探讨之前，请先回顾一下威士忌分析的示例。既然我们已经用了相似性测度来寻找相似的纯麦苏格兰威士忌，为什么又要进一步寻找相似威士忌的簇呢？

原因之一是，我们单纯想进一步了解这个问题。这是一个探索性数据分析示例，包含大量数据的行业应该对此持续投入人力物力，因为这样的探索大有裨益。在本示例中，我们之所以关注苏格兰威士忌，仅仅是因为想知道口味的天然分组——因为我们希望理解该"业务"，而这或许能带来产品或服务质量的提升。假设我们在一个富裕的社区开了一家小店，而本店的经营策略之一是让邻居们知道本店是购买纯麦苏格兰威士忌的好去处。虽然由于场地和库存资金有限，因而店内的威士忌种类并非最全，但是本店可以采取广泛多样、博采众长的收藏策略。如果知道纯麦威士忌的口味的分组方式，我们就能（比如）在每个口味分组中找出最知名的一种和知名度略低的一种，或昂贵的一种和价格较为亲民的一种。而这些都基于对威士忌口味分组的深入了解。

现在，对聚类做更一般性的探讨。本章将介绍两种主要的聚类，同时说明相似性的概念。在此过程中，我们可以检验实际的威士忌聚类。

6.4.2 层次聚类

先举一个非常简单的例子。图 6-6 上半部分的平面（即二维实例空间）上排列着 6 个点 A~F。使用欧几里德距离度量相似性，让平面上距离较近的点相似性较高。编号为 1~5 的圆圈将数据点圈起，表示**簇**。该图展示了“层次”聚类的关键要素。这种方法之所以是**聚类**方法，是因为它根据数据点的相似性对其进行了分组。注意，簇只有被另一个簇包含时才会出现重叠。由于这种结构，因而圆圈实际上代表了聚类的层次结构。最普遍（级别最高）的聚类是一个包含所有数据点的簇，即示例中的簇 5。而级别最低的聚类有 6 个（最小簇），即移除所有圆圈时剩下的 6 个数据点。按照图中编号从大到小移除圆圈之后，我们可以得到一系列不同的聚类方式，每个都会包含数量更多的簇。

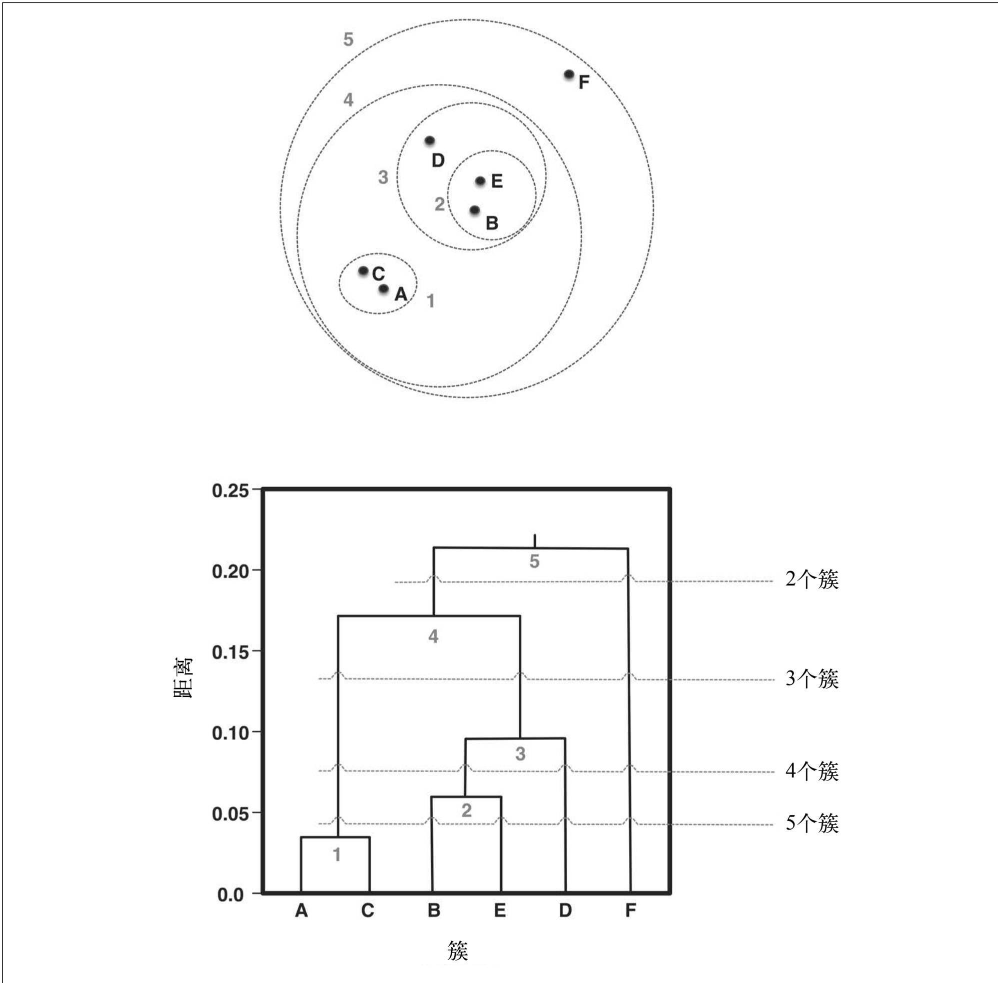


图 6-6：6 个数据点以及它们可能的聚类。上半部分中的 6 个点 A~F 用圆圈 1~5 圈起来，代表基于距离的不同分组。这些分组构成了隐性的层次结构。下半部分是一幅与分组相对应的树状图，明确展示了层次结构

下半部分叫作**树状图**，能够表明簇之间的层次。x 轴表示每个数据点（顺序不分先后，单纯为避免线条交叉）。y 轴则表示簇之间的距离（本章很快会详细探讨这一点）。在该部分最底部（$y = 0$ 时），每个点都是一个独立的簇。随着 y 的增大，不同组的簇开始受到距离限制：先是 A 和 C 被归为一组，再是 B 和 E，然后 BE 和 D，以此类推，直到所有簇都在顶部归为一组。树状图中结合点的数字与上半部分中的圆圈标号相对应。

图 6-6 的两部分都说明，层次聚类不仅仅构造了“聚类”，或是个体的一系列单纯分组，还构造了一系列对数据点进行分组的方法。为明确这一点，不妨设想用一条水平线对树状图进行“切割”，并忽略线以上的部分。随着线逐渐下移，我们会得到包含越来越多个簇的不同聚类方式，如图所示。在名为“2 个簇”的线处切割这幅树状图，线的下方就会出现 2 个组——由 F 独自构成的组和由其余所有数据点构成的组。这次操作相当于移除了上半部分的图中的圆圈 5。如果我们往下走，在名为“3 个簇”的线处切割这棵树，那么线的下方就会出现 3 个组（AC、BED、F）。而与此对应的是在上半部分的图中移除圆圈 5 和 4，这样一来，我们也能得到相同的 3 个簇。这些簇直观易懂：F 依旧单独成组，A 和 C 构成一组，而 B、E 和 D 构成一组。

层次聚类的好处之一是，数据分析师可以在决定获取的簇个数之前看到分组情况，即数据相似性的“格局”。我们可以根据想要的簇的数目，在图表的任意位置进行切割，如图中的水平虚线所示。注意，一旦两个簇在某个水平处合为一组，它们就将在层次更高水平处保持为一组。

层次聚类通常是从各数据点单独成簇开始的。然后这些簇迭代合并，直到最后只剩下一个簇。这样的合并基于相似性，或所选的距离函数。到目前为止，本章已经讨论了实例间的距离。层次聚类需要的是簇之间的距离函数，同时可以而把实例视作最小簇。这有时候也称为**链接**函数。举个例子，链接函数可以定义为“每个簇距离最近的点之间的欧几里得距离”，然后应用于任意两个簇。



树状图

我们通常可以从树状图中得到两种信息。由于 y 轴代表簇之间的距离，因而树状图可以告诉我们天然簇出现的位置。注意，图 6-6 的树状图中，簇 3（约在 0.10 处）和簇 4（约在 0.17 处）的距离相对较远，这意味着将数据划分成 3 组是较好的选择。另外，树状图中的 F 点在极高的距离水平处才与其他点合为一组，这意味着该点与其他点存在差异，可能是一个“离群点”，需要对其进行进一步探究。

层次聚类最著名的运用出自“生命之树”（Sugden 等 , 2003; Pennisi, 2003），这是一幅包含地球上所有生命的层次发展史图。这幅图基于某种 RNA 序列的层次聚类。交互式生命之树的一部分如图 6-7 所示（Letunic & Bork, 2006）。就像此处一样，通常为了节省空间，大型的层次树会采用径向方式展示。该图展示了全基因组测序的全球（分类学）发展史，由 Francesca Ciccarelli 及其同事（2006）自动重构，其中心是地球上所有生命“最后的共同祖先”，由此出现了三个生命分支（真核生物、细菌和古生菌）。图 6-8 放大了树的一部分，其中包含幽门螺杆菌（能导致胃溃疡）。

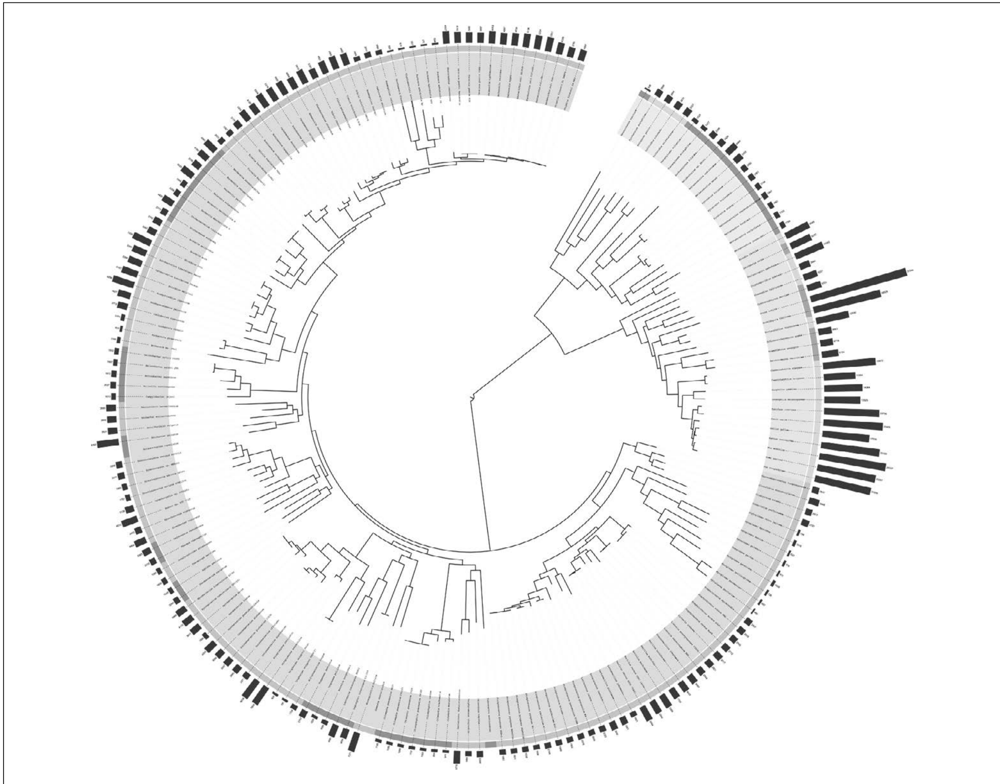


图 6-7：展示生命之树——物种的大型层次聚类——的发展史图，按径向展示

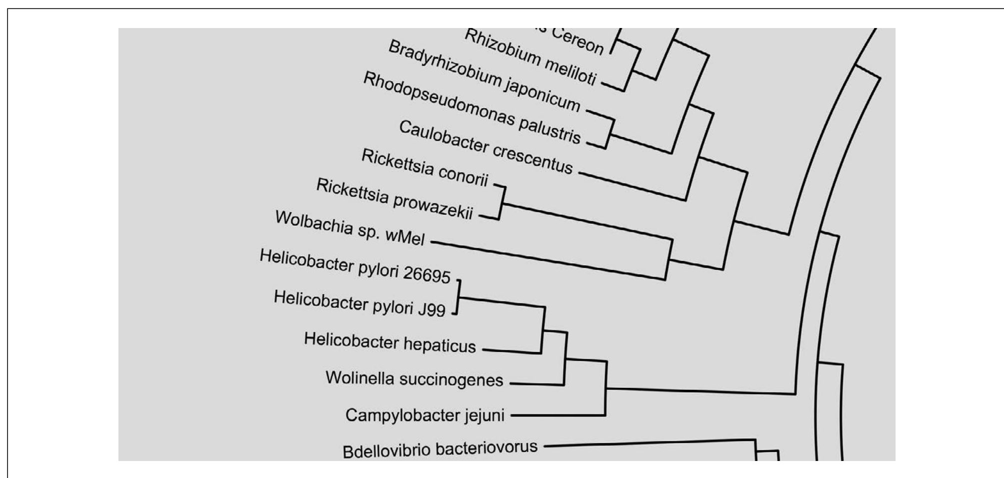


图 6-8：生命之树的一部分[6]

注 6：包含几种生物及其层次关系。——译者注

回到本章开头的示例，图 6-9 是一幅树状图，其上半部分展示了 50 种纯麦苏格兰威士忌用 Lapointe 和 Legendre（1994）发明的方法进行聚类的结果。在对该树状图进行切割后，我们可以根据自己的要求得到任意数量的簇，比如，在移除最上方的 11 个连接分类后，可以得到 12 个簇。

图 6-9 的下半部分则以 Foster 的新欢——Bunnahabhain 威士忌为中心，将部分层级进行了闭合。在 6.2.1 节中，我们找到了与之相似的威士忌，而图 6-9 告诉我们，Bunnahabhain 的这些最近邻（Tullibardine、Glenglassaugh 等）在层次中的确很快与之归为一组。（你可能会感到迷惑，为什么聚类结果与相似性排序不完全一致。这是因为，这五种与 Bunnahabhain 最为相似的威士忌中，可能有的与其他威士忌更为相似，所以它们会在与 Bunnahabhain 合并前，先与那些威士忌并为一类。）

有趣的是，从威士忌分类的角度看，基于口味的纯麦威士忌分组并不与基于苏格兰地区规划的分组（苏格兰威士忌分类的常用基准）完全一致。但 Lapointe 和 Legendre（1994）指出，这两者存在相关性。

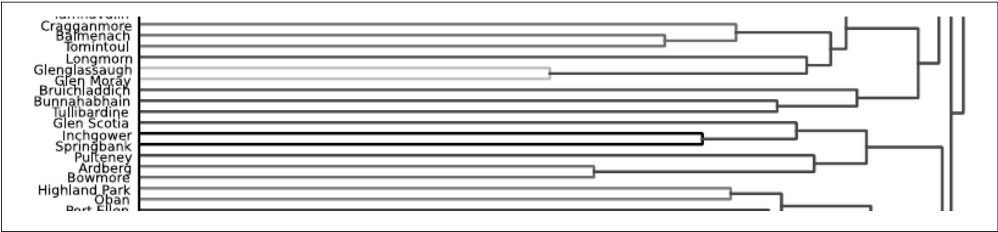


图 6-9：苏格兰威士忌的层次聚类。这一小部分层级展示了 Bunnahabhain 及其近邻的聚类结果

所以该专卖店主不应仅仅储存知名度最高的苏格兰威士忌，或一些高地、低地和艾雷岛品牌的威士忌，而应该在不同的簇中选择库存。或者提供一份帮助纯麦威士忌爱好者挑选威士忌的指南。[7] 例如，因为 Foster 喜欢他的朋友在某天晚餐时推荐给他的 Bunnahabhain，所以他可以从聚类结果中找出其他与之“最相似”的威士忌（Bruichladdich、Tullibardine 等）。数据显示，口味最与众不同的纯麦威士忌是最上面的 Aultmore——它最后才与其他威士忌合为一组。

6.4.3 最近邻回顾：根据形心的聚类

层次聚类关注的是不同实例间的相似性，以及如何依据相似性将它们进行链接。而另一种考虑聚类数据的方法是关注簇本身，即实例构成的组。最常见的后一种方法是用每个簇的“簇中心”，或称**形心**，来代表每一个簇。图 6-10 展示了该理念在二维空间中的应用：此处有 3 个簇，其实例均用圆圈表示。每个簇都有一个形心，用实线星形表示，这颗“星”并不一定是某个实例，而是这个组的几何中心。只要有数值实例空间和度量其中距离的方法（当然，如果是高维空间，我们就无法这么准确地对簇进行可视化），这种理念就可以应用于任意数目维度的空间。

注 7：已经有人完成了，参见 David Wishart（2006）的 *Whisky Classified: Choosing Single Malts by Flavour*。

最常用的基于形心的聚类算法称作 ***k*- 均值聚类**（MacQueen, 1967; Lloyd, 1982; MacKay, 2003），由于该方法在数据科学领域使用频繁，因而值得我们对其主要概念加以探讨。*k*- 均值的“均值”指的是形心，即簇中实例在每个维度上的值的算术平均值（平均值）。因此在图 6-10 中，在确定每个簇的形心的位置时，我们既要对簇中所有实例的 *x* 值求平均，以得到形心的 *x* 坐标，还要对簇中所有数据点的 *y* 值求平均，以得到形心的 *y* 坐标。一般来说，形心是簇中所有实例的每个特征值的平均值。计算结果如图 6-10 所示。

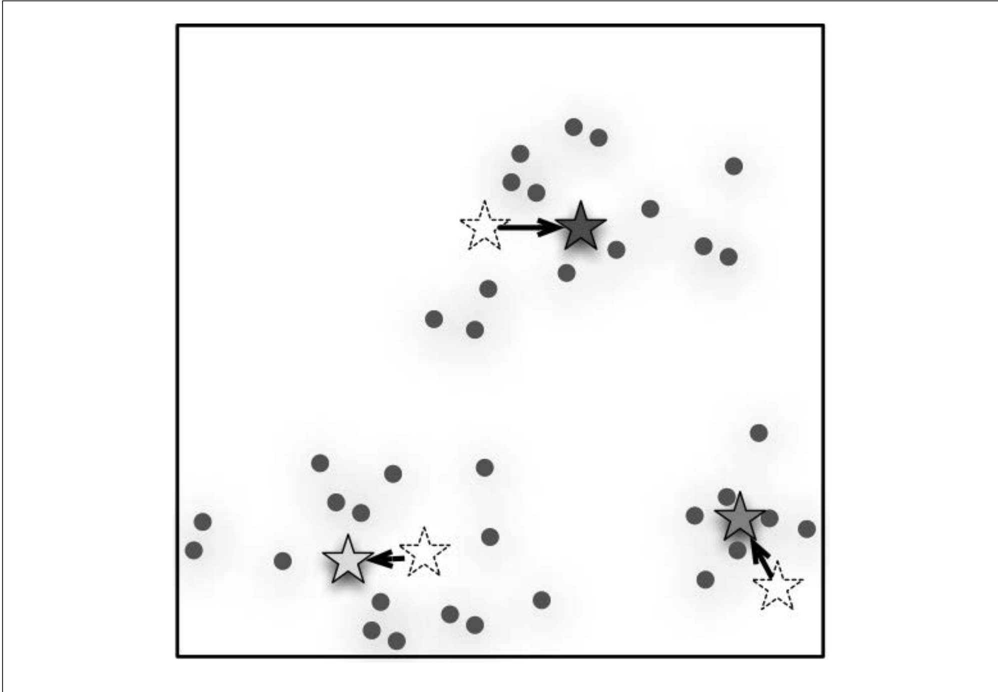


图 6-10：*k*- 均值算法的第二步：找到第一步中发现的簇的实际中心

而 *k*- 均值中的 *k* 仅是在数据中找到的簇的个数。与层次聚类不同，*k*- 均值聚类首先需要确定簇个数 *k*。因此在图 6-11 中，分析师先规定 $k = 3$，然后 *k*- 均值聚类算法才会在聚类算法终止时找到 3 个簇形心（图 6-10 中的 3 个实线星形），进而提供数据点究竟属于哪个簇的信息。这种方法有时也叫作最近邻聚类，因为后一步所提供的恰恰是每个簇包含的所有距离形心最近（而离其他形心相对较远）的点的信息。

k- 均值算法寻找簇的过程简单而巧妙，因此有必要对其进行说明。图 6-11 和图 6-10 就是这种方法的展现，从选定 *k* 个初始簇中心开始，这种选择通常是随机的，但有时也会通过选择实际数据点的其中 *k* 个，或由用户指定，或根据数据预处理结果决定一系列恰当的初始中心。图 6-11 中的星形点就是这些初始中心（$k = 3$），随后算法开始进行。在判定每个数据点与中心的距离关系后，与这些中心相对应的簇就形成了，如图 6-11 所示。

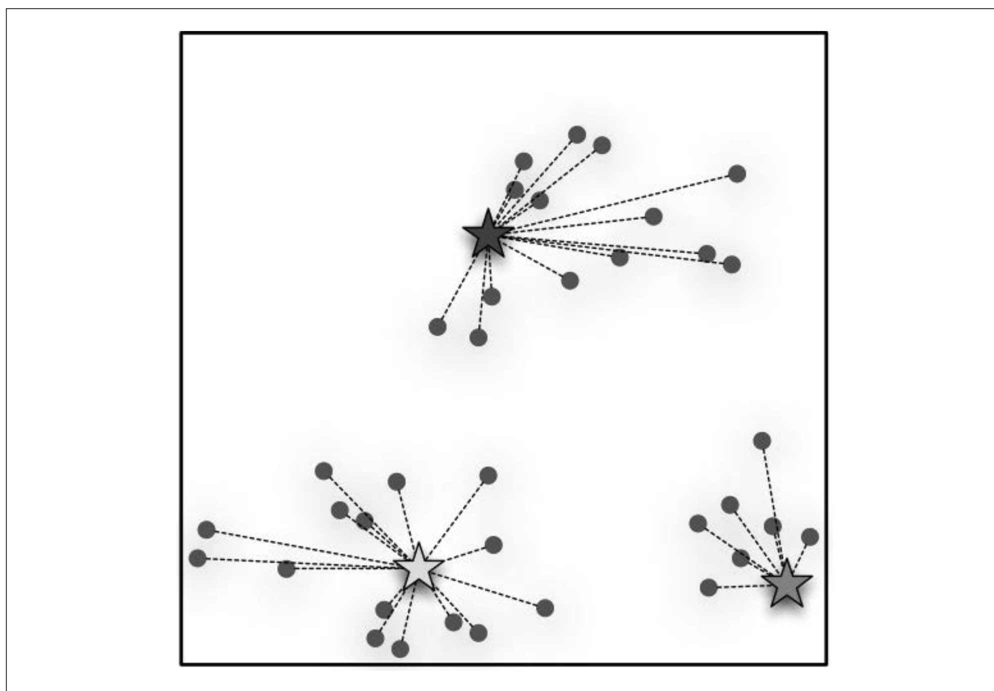


图 6-11：*k*- 均值算法的第一步：找到与所选形心（可能随机选择）距离最近的点，这一步后我们会得到第一组簇

接下来，我们要重新计算每个簇的中心，以找到簇形心的实际位置。由图 6-10 可知，簇的中心通常会发生改变，新的实线星形的确更接近直观上的簇中心，而事实上也大抵如此。随后我们只需不断迭代该过程：由于簇中心发生了变化，因而我们需要重新判定每个数据点的归属（见图 6-11），然后再次计算每个簇中心的位置。直到簇不再发生变化时（或达到某种停止条件时），算法终止。

图 6-12 和图 6-13 展示了对 90 个数据点进行 $k = 3$ 的 *k*- 均值聚类的运行过程。这个数据集更接近现实情况，因为其中不含像前一个示例那样可用肉眼确定的簇。图 6-12 中是聚类之前的初始数据点，而图 6-13 则是在 16 次迭代之后的聚类结果，其中 3 条（不规则）线代表的是每个簇形心从初始（随机）位置到最终位置的移动路径。图中的 3 个簇用不同的符号（●、× 和▶）进行了区分。

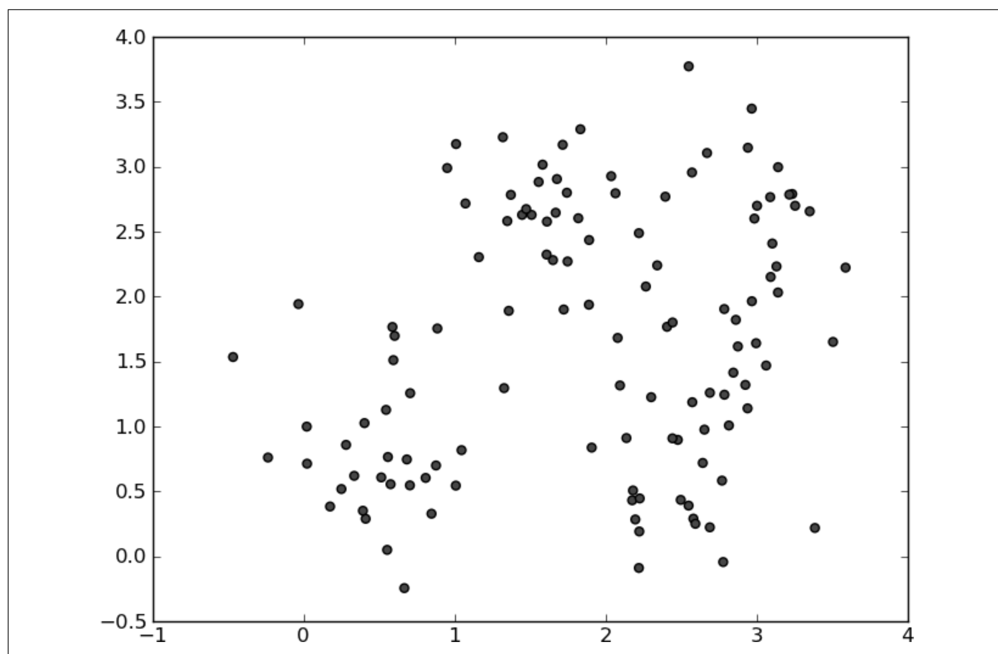


图 6-12：对 90 个数据点做 $k = 3$ 的 k- 均值聚类。图中是数据点的初始状态

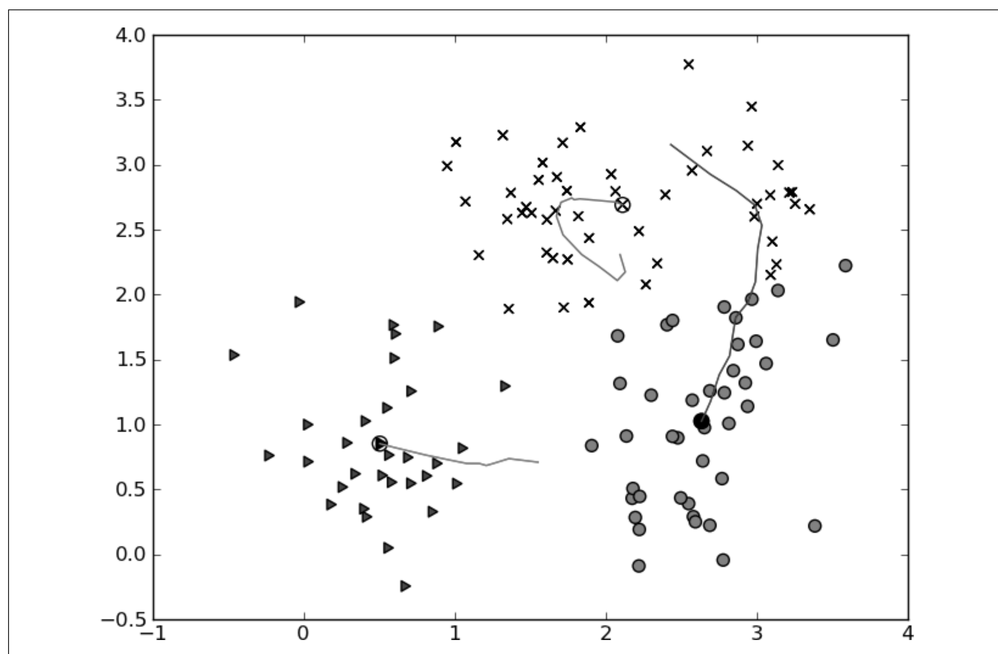


图 6-13：对 90 个数据点做 k=3 的 k- 均值聚类。该图像展示了形心在 16 次迭代过程中的移动路径（3 条线），数据点的符号标记代表该点最终属于的簇

单独一次 k- 均值算法运行不一定能产生好的聚类。运行一次聚类能得到局部最优结果，即局部最佳聚类，但这取决于最初的形心位置。因此我们往往需要多次运行 k- 均值算法，且每次都随机选择不同的初始形心。最后比较聚类结果时，可以通过检验簇来比较（后文将详细讲述），也可以根据数值指标来比较，如簇的**失真度**。失真度，即簇中所有数据点与其对应簇形心的距离平方之和。失真度越低，聚类越优良。

就运行时间而言，k- 均值算法非常高效。由于只需在每次迭代时计算每个数据点到簇中心的距离，因而即使运行多次，该方法也相对较快。而层次聚类则通常较慢，因为它需要在每次迭代时计算每两个簇之间的距离，起初甚至需要计算每两个数据点之间的距离。

k- 均值算法之类的形心算法通常需要注意的一点是如何确定恰当的 k 值。我们可以简单地尝试不同 k 值，以比较哪一个的结果更好。由于 k- 均值通常用于探索性数据挖掘，因而分析师必须检验聚类结果是否有意义，这个过程往往可以帮助确定合适的 k 值。如果有的簇过小且过于详细，那么可以减小 k 值；而如果有的簇过大且过于宽泛，则可以增大 k 值。

我们还可以采用更客观的方法，即不断增大 k 值，并用图像描绘出不同度量方法（有时也称作指标）下不同 k 值对应的聚类的质量。随着 k 的增大，聚类的质量终会趋于稳定。如果该度量方法是最小值最优，则聚类的质量收敛于底部；如果为最大值最优，则聚类的质量收敛于顶部。虽然在决定 k 值时，需要适当加以判断，但我们往往会选择平缓趋势最早出现时的 k 值。维基百科的文章“Determining the number of clusters in a data set”描述了评估簇优良程度的多种方法。

6.4.4　示例：对商业新闻报道进行聚类

接下来是一个基于形心的聚类算法的具体示例：识别新闻聚合器发布的商业新闻报道的天然分组。该示例的目标是简要识别有关某个公司的新闻报道的不同分组。这个示例可能适用于某些具体应用，比如：快速了解一家公司的新闻而无须详细阅读新闻报道；在新闻优先级处理中对现有的报道进行分类；或在进行更重要的数据挖掘工作——如把商业新闻报道与股票行情相联系——前对数据有一个大体了解。

本例选用新闻报道的大型文本集：汤森路透社文本研究集（TRC2）。这是路透社建立的新闻语料库，包含从 2008 年 1 月到 2009 年 2 月（14 个月）的共 1 800 370 篇新闻报道，可供研究人员使用。为使该例在保证真实的前提下易于处理，我们将只提取其中提及苹果公司（股票代号为 AAPL）的新闻。

1. 数据准备

由于在本例中，需要把文本作为数据处理，而该做法前文并未提及，因而有必要先详细讲解一下数据准备工作。第 10 章包含更多挖掘文本的细节，读者不妨一读。

该语料库中，大型企业往往会在作为新闻主题时被提及，这样的新闻包括收益报告、合并公告等。与此同时，它们也常常作为每周业务总结、活跃股票清单和行业重大事件新闻中的次要主题。比如，许多有关个人计算机行业的新闻都提到了惠普和戴尔的股票价格在某天的反应情况，即使这两个公司都与新闻中的事件无关。因此，我们提取了那些标题上明确提及了“苹果公司”的新闻，这意味着该新闻有很大可能是关于苹果公司的新闻。这样的新闻有 312 条，我们将会看到它们涵盖了许多主题。

在聚类之前，需要先对新闻进行基本的网页文本处理：删除其中的 HTML 和 URL 地址、统一单词大小写，以及删除在语料库中出现次数过少（在不超过两篇文章中出现过）和过多（在超过 50% 的文章中出现过）的词，并把剩下的单词编为**词汇表**，以供下一步使用。然后我们用“TFIDF 得分”给文章中的每个单词评分，把文章转化成数值特征向量。TFIDF（词频与逆文档频率的乘积）得分表示在考虑某个词在语料库中的频率的影响后，该词在文章中的频率的得分。本书将在第 10 章对 TFIDF 进行详细讲解。

此处相似性测度用的是余弦相似性，如公式 6-5 所示。该度量方法在文本应用中常用于度量文档的相似性。

2. 新闻报道聚类

我们选择把新闻分为 9 簇（即在 *k*- 均值中，$k = 9$）。下文展示了这些簇的描述信息，以及簇中包含的一些新闻标题。应该记住的是，不仅仅是这些标题，而是整篇新闻报道都被用于进行聚类。

簇 1. 有关等级变化和目标股价调整的分析师公告。

- 加拿大皇家银行（RBC）将苹果公司（AAPL.O）目标价格从 $190 调整为 $200；保持高评级
- THINKPANMURE 给苹果公司买入评级；目标价格 $225
- AMERICAN TECHNOLOGY 将苹果公司（AAPL.O）评级由中性升为买入
- CARIS 将苹果公司（AAPL.O）目标价格从 $170 调整到 $200；评级高于平均
- CARIS 将苹果公司（AAPL.O）目标价格从 $165 调整到 $155；评级保持高于平均

簇 2. 在每天交易过程中及交易结束后，苹果公司股票价格变动的新闻。

- 苹果公司股价收付损失，价格仍下降 5%
- 苹果公司业绩强劲，股价上涨 5%
- iPhone 需求乐观，苹果公司股价上涨
- 苹果公司股价在周二事件前下跌
- 苹果公司股价飙升，投资者爱其估值

簇 3. 2008 年出现了许多关于苹果公司卓越的 CEO——史蒂夫 · 乔布斯及其与胰腺癌抗争的新闻，乔布斯逐渐恶化的健康状况引发了大众的热烈讨论，许多商业新闻都在推测没有乔布斯的苹果公司的未来将会如何，如下：

- 分析——苹果公司的成功不仅与史蒂夫 · 乔布斯有关
- 新闻人物——乔布斯的勇敢和魅力是苹果公司的公众形象
- 专栏——史蒂夫离开后的苹果公司损失了什么：Eric Auchard
- 苹果公司将因乔布斯的健康问题面临诉讼
- 即时观点 1——苹果公司 CEO 乔布斯将请病假
- 分析——没有乔布斯的苹果公司让投资者感到恐惧

簇 4. 苹果公司的公告和新品发布。表面上这些新闻都很类似，但其主题各有不同：

- 苹果公司提出 iPhone“推动了”电邮软件
- 苹果公司 CFO 预估第二季度利润约为 32%

- 苹果公司对 2008 年 iPhone 销售目标大有信心
- 苹果公司 CFO 预估第三季度毛利保持稳定
- 苹果公司将在 3 月 6 日讨论 iPhone 软件计划

簇 5. 其他国家有关 iPhone 和 iPhone 交易的新闻。

- MegaFon 称将在俄罗斯销售苹果 iPhone
- 泰国 True Move 将与苹果公司合作销售 3G iPhone
- 俄罗斯零售商将于 10 月 3 日开始销售苹果 iPhone
- 泰国 AIS 与苹果公司交涉 iPhone 发布日期
- 软银（Softbank）称将在日本销售苹果 iPhone

簇 6. 正常交易时间之外（即开盘前和收盘后）的股价变动。

- 开盘前——苹果公司股价因券商动作缓慢增长
- 开盘前——苹果公司股价上涨 1.6%
- 开盘前——券商评级下调，苹果公司股价下滑
- 收盘后——苹果公司股价下跌
- 收盘后——苹果公司股价继续下跌

簇 7. 该簇无一致主题。

- 分析——别太高兴！苹果公司将面临不确定的 2009 年
- 新闻一瞥——苹果公司 Macworld 大会
- 苹果公司关注纤薄本及线上电影租赁
- 苹果公司乔布斯结束电影计划演讲

簇 8. 有关 iTunes 和苹果公司在数字音乐销售中的地位的新闻。

- 紧跟时代——诺基亚进入数字音乐市场，与苹果公司对抗
- 苹果 iTunes 上升为美国第二大音乐零售商
- 苹果公司或将降低 iTunes 竞争热度
- 诺基亚将接棒苹果，开发音乐触屏手机
- 苹果公司与各品牌协商无限音乐事宜

簇 9. “新闻短讯”是路透社新闻报道的其中一种，通常是几条语言精练的列项短句（如：“• 据称新电影 DVD 发售当天即可在 iTunes 购买”）。“新闻短讯”的内容各异，但因为其形式相似，我们将其归为一组：

- 新闻短讯——苹果公司发布 Safari 3.1
- 新闻短讯——苹果公司推出 ilife 2009
- 新闻短讯——苹果公司宣布 iPhone 2.0 软件测试
- 新闻短讯——新电影 DVD 发售当天将在 iTunes 同步上线
- 新闻短讯——苹果公司称 iPhone 3G 首周销量达一百万

可以看出，有的簇很有趣，而且主题一致，有的却不然，有的仅是表面上相似的文本的集合。统计学中有一句老话："相关性不是因果关系"，指两个事件共现并不意味着两者之间存在因果关系。聚类中也有一句相似的警告："语法相似不等于语义相似"，不能因为两件事物（尤其是两篇文章）有相同的表面特征，就认定它们语义上也一定相关。虽然我们不期望每个簇都有意义且有趣，但聚类往往可以在数据中发掘出出乎意料的结构。簇还能使我们发现崭新且有意思的数据挖掘机会。

6.4.5 理解聚类结果

规定好了数据格式并将其聚类，下一步又该如何？上文中提到过，聚类结果要么是树状图，要么是一系列簇中心及其对应的数据点。那么该如何理解这些聚类结果呢？这一点尤其重要，因为聚类通常用于探索性分析，而探索性分析的关键就是理解是否有什么被发现了，如果是，究竟发现了什么。

对聚类和簇的理解依赖于聚类所用的数据，以及其应用背景，但也存在一些通用的方法。其中几种我们已经运用过了。

请思考上面的威士忌示例。"威士忌研究员"Lapointe 和 Legendre 把聚类树状图切割到剩下 12 簇。以下是其中两个。

A 组

品种：Aberfeldy、Glenugie、Laphroaig、Scapa

H 组

品种：Bruichladdich、Deanston、Fettercairn、Glenfiddich、Glen Mhor、Glen Spey、Glentauchers、Ladyburn、Tobermory

因此，在检验簇时，我们只需观察每个簇内的威士忌。这似乎很容易，但请记住，该示例仅为一个被选入本书的演示范例。这其中到底是哪一点使它对簇的检验相对容易（因而成为本书中一个优良示例）呢？你可能会觉得，这是因为该例中的威士忌总数很少，因此很容易便可以观察它们全部。这一点没错，但并不是问题关键。这是因为即使示例中威士忌种类繁多，我们仍然可以对每个簇中的威士忌取样来展示每个簇的组成。

要理解这些簇，更重要的因素（至少对于那些稍微了解纯麦威士忌的人来说）是簇中的元素可以表示为威士忌的**名字**。本例中，这些数据点的名字本身就具有意义，包含着能被专家理解的信息。

这一点给我们的启发可以推广到其他领域。例如，如果要对某个大型零售商的用户进行聚类，那么用户的姓名可能意义不大，因此这种理解聚类结果的方法也就毫无用处；然而，如果 IBM 要对它的商业用户进行聚类，那么（至少其中的很多）用户的名字就对经理或销售人员意义重大。

如果不能单纯地展示数据点的名称，或展示名称意义不大，又应该怎么做呢？请再回顾一下威士忌示例里的聚类，但是这次要多观察里面的一些信息。

A 组

- 品种：Aberfeldy、Glenugie、Laphroaig、Scapa
- 簇内最佳：Laphroaig（Islay 品种），10 年，86 分
- 一般特征：全金；果香、咸香；中等；油滑、咸味、雪利酒味；苦味

H 组

- 品种：Bruichladdich、Deanston、Fettercairn、Glenfiddich、Glen Mhor、Glen Spey、Glentauchers、Ladyburn、Tobermory
- 簇内最佳：Bruichladdich（Islay 品种），10 年，76 分
- 一般特征：白葡萄酒、淡色；甜香；柔滑、轻盈；甜味、苦味、果味、烟熏味；苦味、轻盈

这里出现了两条有助于理解聚类结果的额外的信息。其一是，在簇成员之外，这里还列出了一个“样例”成员，即“簇内最佳”威士忌。该威士忌由 Jackson（1989）评出（这条额外信息不纳入聚类算法中）。我们还可以列出簇内最出名或最畅销的一种威士忌。这个方法在簇内成员过多时尤其有用，因为谨慎挑选这些“样例”比随机抽样更有说服力。然而，这依旧建立在实例的名称有意义的基础上。另一个对商业新闻报道聚类的示例，则对这种总体思路略微做了改动：展示“样例”新闻及其大标题，因为这些大标题是新闻的有意义的摘要。

本例还阐述了另一种理解聚类结果的方法：簇成员的平均特征，即簇形心。任何聚类过程都可以应用这种展示形心的方法。但这样是否有意义，则取决于数据的值本身是否有意义。

6.4.6 *用有监督学习产生簇描述



前方有技术细节！

这一节描述了一种自动生成簇描述的方法。这种方法比之前讨论的那种更为复杂。它涉及将无监督学习（即聚类）和有监督学习相结合，以创造出一种簇的差异描述。如果你是第一次了解聚类和无监督学习，那么你可能会在阅读中产生不少疑惑，因此我们将这一节加了星标（高级学习资料）。即便你跳过不读也不影响前后文的连续性。

无论产生聚类的方式如何，最终我们都会知道每个数据点被分配到哪个簇。而簇形心实际上描述了簇成员的平均水平。问题是，虽然这种描述可能会十分详尽，但是我们无法从中了解簇之间的差别。我们想知道：**到底是什么因素将每个簇区分开来**？而这正是有监督学习方法的用处，因而我们可以运用这种方法。

运用该方法的一般步骤为：首先给每个实例添加簇标签，而该标签也能作为类标签；然后对有标签的实例集应用有监督学习算法，以产生每个类（或簇）的分类器；其后通过观察分类器的描述，（很可能）得到相应簇的易于理解而又具体的描述。重要的是，这些正是**差异**描述，回答了“到底是什么因素将每个簇区分开来”。

从现在起，本节将把簇和类等同，并无差别地混用两者。

原则上我们可以使用任何预测（有监督）学习方法来生成簇的差异描述，但此处重要的应是**易理解性**。因为我们要把学习后的分类器定义作为簇描述，所以需要一个能达到该目的的模型。因为 3.4 节展示了如何从分类树中提取规则，所以可以选用这个方法。

建立分类任务的方法有两种。由于我们有 k 个簇，因而可以建立一个 k- 类任务（一个类对应一个簇）。或者，我们也可以分别建立 k 个学习任务，每个任务都用于将 1 个簇与其他（$k-1$）个簇区分开。

本节将用第二种方法来解决威士忌聚类问题，按照 Lapointe 和 Legendre 的分配簇的方法（详见“A Classification of Pure Malt Scotch Whiskies”附录 A），将威士忌分为 12 个簇，标号为 A~L。我们将在原数据中增加一列簇归属信息，表明每种威士忌属于哪一簇。然后利用二分法，轮流用每个簇与剩余簇进行分类。这里我们选了 J 组，Lapointe 和 Legendre 是下面这样描述的。

J 组

- 品种：Glen Albyn、Glengoyne、Glen Grant、Glenlossie、Linkwood、North Port、Saint Magdalene、Tamdhu
- 簇内最佳：Linkwood（Speyside 品种），12 年，83 分
- 一般特征：全金；苦香、泥煤香、雪利酒香；轻盈到中等、圆润；甜味；苦味

你可以回顾 6.2.1 节中各种威士忌的 68 个二元特征。现在数据集有了标签（J 或 not_J），以标明该威士忌是否属于 J 组。下面是数据集的一部分：

```
0,0,0,...,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,J     % Glen Grant
0,0,0,...,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,not_J % Glen Keith
0,0,0,...,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,0,0,not_J % Glen Mhor
```

“%”后面的文本是威士忌名称的注释。

然后我们把该数据集输入到分类树学习器中[8]，结果如图 6-14 所示。

注 8：特指 Weka 的 J48 过程（http://www.cs.waikato.ac.nz/ml/weka/），但不包括剪枝。

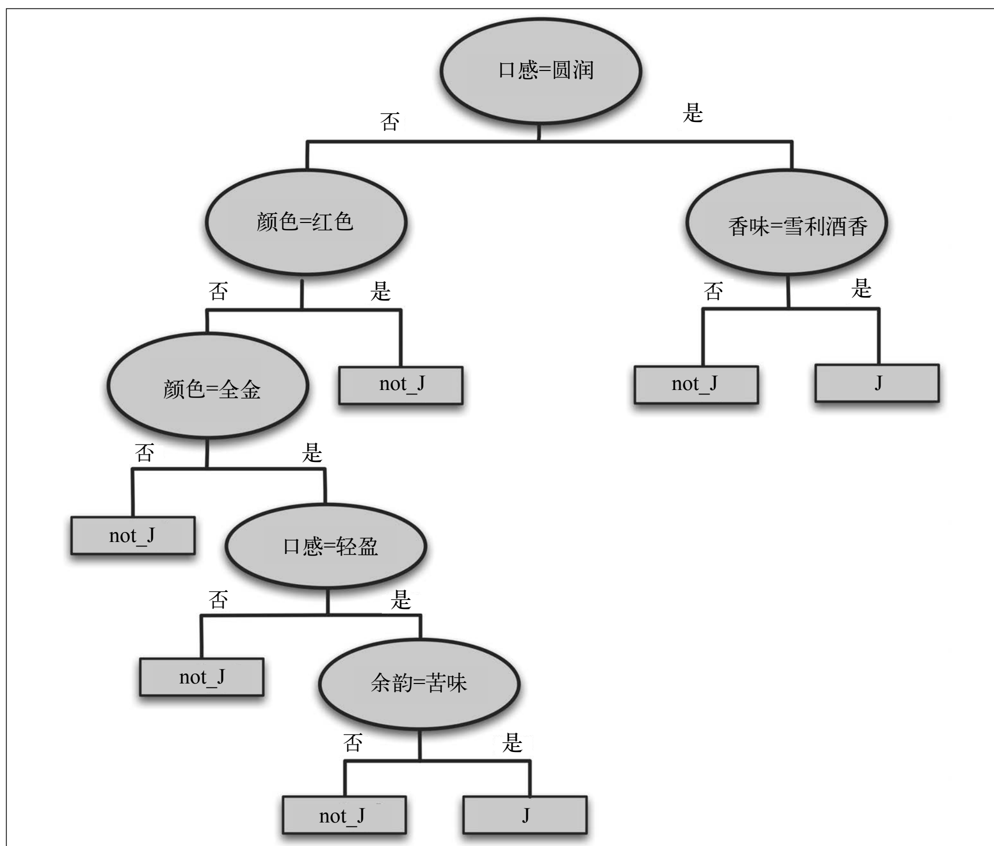


图 6-14：根据 J 组数据，对所有威士忌数据进行学习得到的决策树。最右边的叶节点对应口感圆润、带有雪利酒香的威士忌，该分类下的威士忌大多数来自 J 组

我们只关注这棵树中标签为 J 的叶节点（忽略标签为 not_J 的叶节点），这样的叶节点只有两个。根据根节点到叶节点的路径，我们可以提取出两条规则：

(1)（口感 = 圆润）且（香味 = 雪利酒香 = 1）⇒ J

(2)（口感 = 圆润）且（颜色 = 红色）且（颜色 = 全金）且（口感 = 轻盈）且（余韵 = 苦味）⇒ J

粗略地将上文翻译成自然语言，则 J 组的威士忌的特征为以下两者之一：

(1) 口感圆润，带有雪利酒香；

(2) 颜色为全金（但不是红色），口感轻盈（却不圆润），带有苦味的余韵。

这样的描述是否比上文中 Lapointe 和 Legendre 提供的描述更好呢？这要看你喜欢哪种，但你要知道，这两种描述**类型**不同。Lapointe 和 Legendre 的描述是**特性**描述，描述的是簇的典型特征，而不管其他簇是否也有同样的特征；决策树生成的描述是**差异**描述，只描述该簇与其他簇不同的特征，而忽略簇内成员共有的特征。换句话说，特性描述关注的是组内

共性，而差异描述则关注的是组间差异。两种方式没有哪个绝对更好，具体要取决于你的使用目的。

6.5　退一步：解决业务问题与数据探索

在看过许多将数据科学的基本概念付诸实践的示例后，你可能已经明白，即使目标相似，聚类问题与预测建模问题也或多或少存在着不同。我们来探索一下其中的原因。

在预测建模示例和直接运用相似性的示例中，所关注的都是解决特定的商业问题。前文曾强调过，数据科学的基本理念之一就是要尽可能精确地定义数据挖掘任务的目标。还记得 CRISP 数据挖掘流程吗？图 6-15 再次展示了它的流程图。在商业理解 / 数据理解的小循环中，我们应该用尽可能多的时间来给需要解决的问题下具体而准确的定义。在预测建模的应用中，我们需要具体地定义目标变量，而在第 7 章我们会了解到，随着对数据科学理解的加深，对问题的定义也会愈发具体。在相似性匹配示例中，同样有对目标的具体描述——找到相似企业来使工作的结果最优，因而我们需要具体定义“相似”的含义。如果想找到相似的威士忌，尤其是味道相似的威士忌，那么我们仍需要收集数据和表示数据以便于精确地找到它们。后文将探讨如何运用数据科学的框架，将商业问题分解为多个定义明确的部分，然后运用数据科学方法来一一解决。

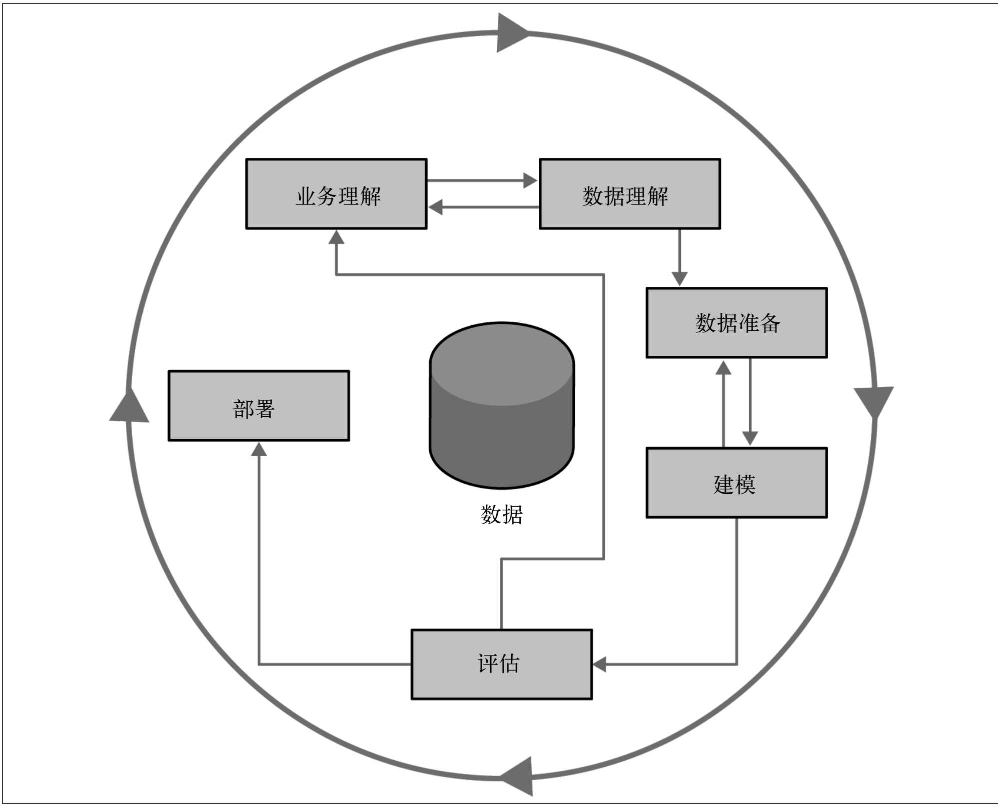


图 6-15：CRISP 数据挖掘流程

然而，并非所有问题都如此定义明确。如果在业务理解环节得出如“我们虽然不太确定需要解决的问题，但仍希望探索一下数据”这样的结论，该怎么做？需要应用聚类方法解决的问题往往属于此类。此时要运用**无监督**分类，找到“天然”存在的分组（当然，这依赖于相似性测度的定义）。

为便于讨论，我们先将问题简化：分为有监督部分（如预测建模）和无监督部分（如聚类）。当然数据科学领域并非那么呆板，之前介绍过的数据挖掘技术几乎都能用于数据探索。但如果简要把问题划分成有监督和无监督两部分，对问题的探讨就会更加清晰。在数据挖掘流程中，我们需要权衡在哪个环节、以何种方式投入精力。对有监督问题而言，由于已经花了很大功夫具体定义问题，因而到了数据挖掘流程中的评估环节，我们就已经有了清晰的评估指标——建模结果是否能够解决所定义的问题。举个例子，如果已经确定，目标是改善在合约即将到期时用户的流失情况，那么我们就可以评估模型是否满足了这个要求。

与之相比，无监督问题往往更具探索性。我们可能意识到，如果能对企业、新闻报道或威士忌聚类的话，就能更加了解我们的任务，从而能够对某些方面做出改进。然而，我们可能没有目标的具体公式。如果要求过于具体和准确，就有可能发现不了数据中的重点。但这两者需要权衡，即如果在数据挖掘流程前期没有对问题做出具体定义，那么在后期的评估环节就需要花更多时间。

特别是对于聚类问题而言，理解其结果所揭示的信息（如果存在）往往很困难。即使聚类结果似乎揭露了有趣的信息，我们也经常不清楚如何用该结果来优化决策。因此，我们必须把额外的创造力和商业知识运用到聚类问题的数据挖掘的评估环节中来。

Ira Haimowitz 和 Henry Schwartz（1997）对新的信贷客户设置信用额度的示例展示了如何用聚类辅助决策。他们基于信用卡数目、账单偿还情况和给公司产生的利润，对 GE Capital 的现存客户进行了聚类。最终他们确定了 5 个簇，代表了 5 种非常不同的客户信贷行为（比如，同样消费很多，有的人每月都能如数还清，而有的人账户余额却一直接近信贷限额）。这些不同的客户所适用的信贷额度也大不相同（上文中的两种人里，我们更需要关注后者，以防其出现信贷违约）。但直接将聚类用于该决策的问题是，这些数据在起初设置信用额度时无法获得。因此，简而言之，Haimowitz 和 Schwartz 在获取这些新信息后，又重新开始了整个数据挖掘流程。它们用这些新信息定义了一个精确的预测建模问题：用信贷审批时可获取的数据预测客户对每个簇的归属概率。接下来，这个预测模型可以用于改进初次信用额度设置的决策。

6.6 小结

数据项之间相似性的概念贯穿于整个数据挖掘。本章首先讨论了相似性的广泛应用，从基于数据描述寻找相似个体（或对象），到预测建模，再到聚类。本章探讨了这些用途，并用一些示例来进行了说明。

两个个体之间相似性的一个常用的替代指标是，两者在由特征向量定义的实例空间中的距离。本章对相似性的计算方式和距离的计算方式分别做了一般性介绍和技术细节讲解。另外，本章还介绍了最近邻方法，即通过计算新数据和一些训练数据（目标变量值已知）的

相似性来进行预测工作的方法。在获取一系列最近邻（即最相似的实例）后，我们就能用它们来解决许多数据挖掘问题，如分类、回归、实例评分等。最后，本章揭示了相似性这一基本概念同样也是无监督数据挖掘最常用的方法——聚类——的基础。

本章还探讨了另一个重要概念，它可以用在探索性更强的数据分析方法中。在探索数据时，尤其在使用无监督方法的情况下，我们往往会在数据挖掘流程的业务理解环节花较少的时间，而在评估环节和迭代该循环的过程上花较多的时间。为了方便说明，本章探讨了理解聚类结果的多种方法。

第7章

决策分析思维（一）：如何评估一个模型

基本概念：仔细思考希望从数据挖掘的结果中获得什么；期望值：一个关键的评估框架；思考什么是恰当的对比基线

示例方法：各种评估指标；成本收益估计；期望利润的计算；创建比较基线的方法

请回想一下第5章的开头：作为MegaTelCo的一名经理，你想评估本公司的模型是否真的是一个“好”模型。除了过拟合之外，你还应做何种度量呢？

为了让数据科学给实际应用增加价值，数据科学家和其他利益相关者必须仔细地考虑他们究竟希望通过挖掘数据实现什么。虽然这点听起来像是老生常谈，但令人惊讶的是，它经常被忽略。数据科学家及他们的合作者会经常回避——也许他们甚至都没有意识到——将数据挖掘的结果与他们的实际目标联系起来。其影响既可能表现为统计报告中缺少对统计数据正确性的明确解释，也可能表现为不能找出有意义的方法来测量性能。

但是，我们也应谨慎地对待这类批评。直接测量最终目标通常是非常困难的，原因可能是系统存在缺陷，收集高质量数据的成本太高，或者评估数据与目标变量之间的因果关系很困难。因此，我们需要测量一些有用的替代变量。尽管如此，至关重要的仍然是要考虑清楚究竟要测量什么。即使必须选择替代变量，也要通过严谨的数据分析来实现。

本章面临的最大挑战是，每个应用场景都是不同的，我们无法为分类问题、回归问题或者可能遇到的其他任何问题提供单一且“正确”的评估指标。尽管如此，在对模型进行评估的过程中存在很多共同的主题和争论点，而对于解决这些问题，也存在一些共同的技术和框架。

本书将逐一讨论关于分类（在本章中）、实例评分（例如，根据消费者响应的可能性对消费者进行排序）和类概率估计（在下一章中）等任务的一些评估框架和度量指标。每一项具体的技术都应该被看作对应用场景中不同需求进行深入思考的示例。幸运的是，这些技术的适用范围的确很广。同时，本章也会给出一个通用的框架，用于模型评估和期望值计算，而这个框架可以涵盖各种各样的应用场景。正如将在后面的章节中展示的那样，它也可以作为数据分析式思维的体系化工具，用来指导问题的标准化。

7.1 对分类器的评估

分类模型是一个用来预测类别未知的实例的模型。现在，试想一个二分类模型，其中的类别值按惯常方法简称为“正”（样本）和“负”（样本）。对于这种模型，应如何评估其性能呢？第 5 章讨论了一种评估方法，即将数据集二分为训练集和测试集来评估模型的泛化能力。但是具体应该如何操作呢？

坏的正样本与无害的负样本

在讨论分类器时，我们经常将产生负面效果的样本看作“正”的，而将正常或好的样本看作“负”的。鉴于对“正”和“负”的日常定义，这样的表述对你来说可能会很奇怪。例如：为什么欺诈事件被认为是正的，而正常事件被认为是负的？事实上，这样的措辞在许多专业领域都很常见，包括机器学习领域和数据挖掘领域，在本书中也会这样使用。下面的解释或许能帮助你更好地理解这个问题。

通常，我们用正向结果代表值得关注或**警惕**的事情，而将负向结果看作不值得关注的事情或良性事件。例如，检测生物样本的医学测试（一种分类器），通过检测样本的某些方面来判断是否有疾病。如果检测结果为阳性（也就是正向），则表示存在异常状况；如果检测结果为阴性（也就是负向），则表示并没有什么值得警惕的因素，通常不需要治疗。同样，如果欺诈检测模型检测到用户账户的异常活动，并引发风险预警，则称为正向反馈。虽然负向反馈（只出现合法活动的账户）或许是好的事情，但从欺诈检测的角度来看，它们并不值得关注。

其实保持这种惯用的规则往往是很有意义的，因为我们不必在引入每个领域的时候，重新定义“正”和“负”的含义。你可以将分类器看作一个通过筛选一个主要由（不值得关注的）负样本构成的总体来寻找少数正样本的工具。按照惯例，正样本通常都占少数，至少比负样本要少。因此，尽管对负样本判断错误（**假阳性错误**）的情况可能更多，然而对每个正样本判断错误（**假阴性错误**）的**成本**会更高。

7.1.1 简单准确率的问题

到目前为止，本书一直假设可以使用一些简单的度量标准，比如分类器的错误率或准确率，来衡量模型的性能。

分类准确率是一个常用的指标，因为它很容易测量。但是很遗憾，它对于数据挖掘技术在实际业务问题中的应用来说，还是过于简单了。本章将仔细讨论分类准确率这个指标，以

及它的一些替代指标。

术语“分类器准确率”，在非正式的情况下，可以指对广义上任何一种分类器的性能的测量。在这里，对于**准确率**一词，本书取其技术上的特定含义，即正确决策所占比例，即：

$$准确率=\frac{正确决策数}{决策总数}$$

其也等于 1- **错误率**。准确率是一项在数据挖掘研究中很常见的评估指标，因为它可以用单一的数字来评估分类器性能，而且很容易测量。但是，它过于简单，且会导致一些很常见的问题（Provost, Fawcett & Kohavi, 1998）。为了理解这些问题，需要一种方法来分解和计算分类器导致的不同类型的决策错误。为此本章引入“混淆矩阵”。

7.1.2 混淆矩阵

想要正确地评估分类器，理解**类混淆**和**混淆矩阵**这两个概念是非常重要的，其中后者是一种列联表。涉及 n 类问题的混淆矩阵是一个 $n \times n$ 矩阵，矩阵的每一列表示对应样本的真实类别，而每一行表示预测类别。测试集中的每个实例都有一个真实的类别和一个分类器预测的类别（预测类），它们的组合构成了矩阵的各个单元。简单起见，本章仅讨论一个二分类问题的 2×2 的混淆矩阵。

混淆矩阵可以将分类器做出的决策区分开，明确地展示出一个类别是如何与另一个类别混淆的。通过这样的方式，我们可以单独处理不同类型的错误。首先，要用不同的符号来区分真实的类和模型预测的类。在这里，我们会考虑二分类问题，将真实的类别表示为 p（positive，正向）和 n（negative，负向），将模型预测的类（“预测”类）表示为 Y（Yes，是）和 N（No，否）（就好像模型在说“**是的**，它是正向的”或“**不**，它不是正向的”）。

在如表 7-1 所示的混淆矩阵中，主对角线上的单元包含正确的预测。而分类器中错误的预测是**假正**（被分类器预测为正的负样本）和**假负**（被分类器预测为负的正样本）。

表7-1：一个2 × 2的混淆矩阵，它显示了模型正确的预测（主对角线）和错误的预测（次对角线）

	p	n
Y	真正	假正
N	假负	真负

7.1.3 样本类别不均衡的问题

举一个例子来说明我们需要仔细考虑模型评估。在分类问题中，其中一个类的样本量非常小，这在实际应用场景中是很常见的，因为分类器通常被用于筛选大量正常的、不值得关注的样本，以寻找相对少量的异常样本。比如，寻找遭受欺诈的用户，检查装配线上是否存在缺陷部件，或检测目标消费者实际是否会对营销活动做出响应。因为异常的、值得关注的样本在总体中所占数量通常是很少的，所以我们往往会遇到样本分布不均衡或分布偏斜的情况（Ezawa, Singh & Norton, 1996; Fawcett & Provost, 1996; Japkowicz & Stephen, 2002）。

不幸的是，随着样本偏度（样本类别分布不均衡的程度）的增加，基于准确率的评估方法就会逐渐失效。如果一个样本总体中的类以 999：1 的比例出现，那么只要遵循一个简单的规则——总是选择数量多的类别——就可以获得 99.9% 的准确率。但是如果要寻求非平凡的解决方案，那么这种方法的效果可能并不令人满意。在欺诈检测中，正负样本 1：100 的比例是很常见的；而在其他应用场景中，有的样本偏度甚至超过了 1：106（Clearwater & Stern, 1991; Attenberg & Provost, 2010）。第 5 章提到了类的"基础比率"这个概念，它表示当分类模型将全部实例预测为某一类时，这个模型的性能如何。对于这种偏度很高的样本总体而言，占主导地位的那一类的基础比率可能会非常高，因此在准确率为 99.9%的情况下，这个指标可能无法告诉我们数据挖掘真正实现了什么。

即使当样本偏度不是那么大时，如果样本总体中一个类别比另一个类别更占主导地位，那么准确率也容易变得不准确。请再次回到手机用户流失的那个示例。假设你是 MegaTelCo 的经理，而我这个分析师告诉你用户流失预测模型的准确率是 80%。这听起来很不错，但果真如此吗？我的同事说她的模型准确率只有 64%。这似乎很糟糕，其实是这样吗？

你可能会说：等等，我们还需要更多信息。你这样做是完全正确的（并且这意味着你已经进入了数据分析的思维模式）。那么还需要什么呢？考虑到本节迄今为止所讨论的内容，你可能会很确定地说：需要知道总体中流失用户的比例是多少。假设用户流失的基础比率约为每月 10%，因此如果把流失用户看作正样本，那么在客户群中，预期的正负样本比例为 1：9。因此，只要把所有的用户都看作正常用户（负样本），就可以实现 90% 的准确率！

随着挖掘工作的深入，你又发现我的同事和我其实是在两个数据集上进行了评估。这一点也不奇怪，如果没有事先协调好数据分析工作的话，就会出现这样的情况。我的同事从样本总体中提取代表性样本来计算准确率（保留了原始样本的分布），而我则是创建了用于训练和测试的人工平衡数据集（两种都是常见做法）。现在我同事的模型看起来非常糟糕——她应该可以达到 90% 的准确率，却只有 64%。然而当将她的模型在我的平衡数据集上检验的时候，却得到了 80% 的准确率。这真令人困惑。

最重要的是，准确率这个指标其实存在局限性。在这个编造的示例中，我同事的模型（模型 A）正确识别出了所有的正样本，但只找到 60% 的负样本，最后在平衡数据集上总体达到了 80% 的准确率。相反，我的模型（模型 B）正确识别出了所有的负样本，但只识别出 60% 的正样本。

让我们使用混淆矩阵来更仔细地研究一下这两个模型。这是一个有 1000 名用户的训练总体，它的混淆矩阵的分布如表 7-2 和表 7-3 所示（模型的预测类别分别表示为 Y 和 N）。

表7-2：混淆矩阵A

	流失用户	续约用户
Y	500	200
N	0	300

表7-3：混淆矩阵B

	流失用户	续约用户
Y	300	0
N	200	500

图 7-1 展示了两个模型在平衡样本中和代表性样本中的分类预测情况。正如前面所提到的，两个模型都对 80%的平衡样本进行了正确分类，但是，混淆矩阵的结果表明它们的分类方式是非常不同的。分类器 A 经常错把续约用户预测为流失用户，而分类器 B 会犯相反的错误，把流失用户预测为续约用户。而把两个模型放在代表性样本（保留原始样本分布）上进行测试时，模型 A 的准确率下降到 64%，模型 B 则上升到了 96%。这是一个很显著的变化。那么究竟哪个模型更好呢？

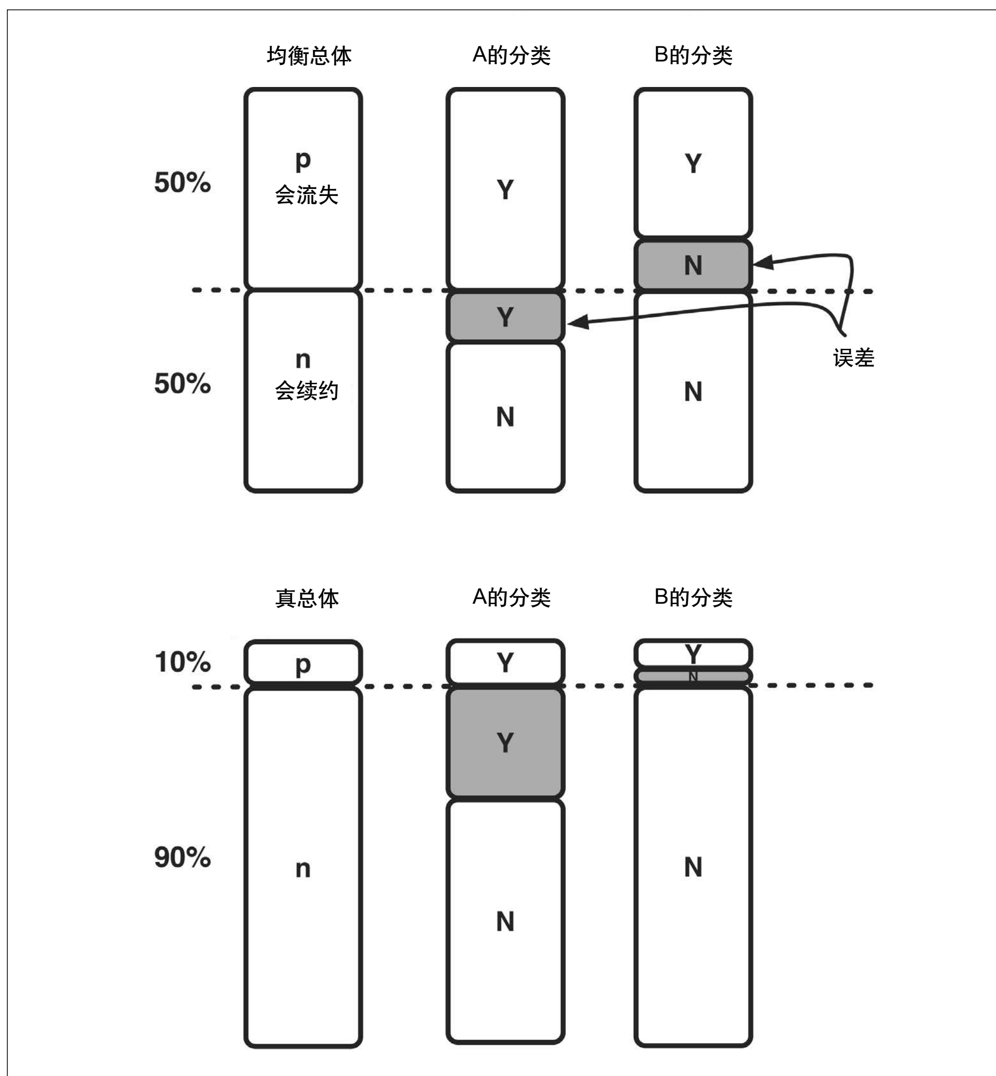


图 7-1：一个说明准确率具有误导性的例子。上面两个流失预测模型，模型 A 和模型 B，在平衡数据集中，产生相同数量的错误（阴影区），但其错误类型不同。因此，当样本比例发生变化时，它们各自的性能也会发生很大变化

我的模型（模型 B）现在看起来似乎比模型 A 更好，因为模型 B 在受关注的总体中有更好的性能（流失用户与续约用户比例为 1∶9）。但是，我们仍然不能下结论，因为准确率这个指标还存在一个问题：不知道如何度量这些不同的错误及正确决策对我们的重要性。这个问题将在下一节讨论。

7.1.4 成本收益不均衡的问题

用分类准确率作为度量指标的另外一个问题是，它不区分假阳性错误和假阴性错误，而默认这两个错误同样重要。而这通常并不适用于真实的应用场景。不同类型的错误会产生不同的成本，因为不同分类方法造成的后果的严重程度不同。

比如在医疗诊断场景中，一个没有得癌症的人被诊断为癌症患者，就是一个假阳性错误。结果可能是患者将接受进一步的检查，最终发现癌症的初步诊断错误。尽管这个错误让患者承受了压力和巨额花费，且造成了很多麻烦，但它不会危及生命。比较一下相反的情况：错误地告诉一个癌症患者没有得癌症，这是假阴性错误。这类错误意味着患有癌症的人会错过早期检测，这可能会产生非常严重的后果。如此看来，这两类错误造成的后果其实非常不同，我们应该分开计算，而它们的成本也应该不同。

回到手机用户流失的示例中，情况则是尽管给了某用户一些用于促使其续约的优惠，然而他还是流失了（假阳性错误）。与之相对的是，因为没有给某用户优惠，所以他流失了（假阴性错误）。无论你决定为每种错误花费怎样的成本，它们都是不一样的；而不管怎样，这些错误都应该被分别计算。

实际情况中，很难想象一个决策者可以对其是犯了假阳性错误还是假阴性错误漠不关心。理想情况下，我们应该仔细评估分类器所做的每个决策的成本或收益。它们合起来，即**期望利润**（或**期望成本**或**期望收益**）。

7.2 分类问题的推广

我们一直在使用分类建模讨论许多具体的数据科学问题，这些问题大部分不仅仅适用于分类问题的范畴。

总体原则是，在将数据科学投入到实际应用时，至关重要的是把关注点放到问题本身：在应用场景中什么是重要的？目标是什么？是否能根据实际目标来评估数据挖掘的结果？

这里举另外一个例子。请把上述思想应用到回归模型而不是分类模型中去。假如我们的数据科学团队计划构建一个电影推荐模型。它可以预测某给定用户对特定电影的喜爱程度，从而给用户提供个性化的推荐。比如每个用户通过给出一到五星的分数来给电影评级，而推荐模型可以据此预测出用户对他们尚未观看的电影的评分。其中一位分析师在评估模型的时候，使用了模型的均方误差（或均方根误差，或 R^2，或其他指标）。我们或许就会问：什么的均方误差？分析师回复：目标变量值的，就是用户给电影评的星数。为什么预测结果的均方误差适合评估该推荐模型？这个指标有意义吗？有没有更好的指标？真希望分析师们仔细考虑过这些问题，但是通常情况下，你会发现他们并没有，而只是在照搬他们从学校课程中学到的方法。

7.3　一个重要的分析框架：期望值

现在我们准备讨论数据分析思维中一个辅助性的通用指标：期望值。期望值的计算过程提供了一个框架，而该框架对于如何思考数据分析问题非常有用。具体地说，它将数据分析思维分解为三个部分：问题的结构、可从数据中提取的分析要素和需要从其他来源获取的分析要素（例如商业知识和专业领域的知识）。

在计算期望值的时候，某种情况下的各种可能结果首先被列举出来。而期望值就是不同结果的加权平均值，其中给予每个结果的权重则是它发生的概率。例如，如果不同的结果代表了不同的利润水平，那么在计算期望利润的时候，可能性高的利润水平会被赋予较高的权重，而可能性低的利润水平则被赋予较低的权重。在本书中，我们假设所要考虑的都是重复任务（比如针对大量消费者，或诊断大量问题），而目标就是实现期望利润的最大化。[1]

期望值框架为分析师的思考提供了一个架构，期望值的计算见公式 7-1。

公式 7-1：期望值计算的一般形式

$$EV = p(o_1)\cdot v(o_1) + p(o_2)\cdot v(o_2) + p(o_3)\cdot v(o_3) + \cdots$$

每个 o_i 都是一个可能的决策结果，$p(o_i)$ 是其发生的概率，而 $v(o_i)$ 是其值。概率值通常可以从数据中获得，但商业价值通常需要从其他来源获得。正如第 11 章将要提到的，数据驱动的建模可能有助于评估商业价值，但这些值通常必须从其他领域获得。

我们将在两个数据科学场景中说明期望值作为分析框架的作用。这两种情况事实上经常被混淆，因此有必要加以明确的区分。为此，请你回顾一下第 2 章中**模型的挖掘**（或归纳）和**模型的使用**之间的差异。

7.3.1　用期望值规范分类器的使用

使用模型时，在很多情景下需要预测一个类别。例如，在目标市场营销中，我们希望把消费者划分为**可能响应用户**和**不可能响应用户**，然后对可能响应用户进行有针对性的营销。但非常不幸的是，每个消费者的响应概率可能都非常低——可能仅有一两个百分点——因此没有一个消费者看起来像可能响应用户。如果按照“常识”中的阈值 50% 来划分用户，那么我们可能找不到任何目标。而许多缺乏经验的数据工作者在发现有的模型把每个人都归为不可能响应用户时，或许会大感意外。

然而，期望值框架可以帮助我们看到问题的症结所在。继续思考目标市场营销的示例。[2] 我们计划为一种产品设计一种促销活动，为了简单起见，这种产品只能通过该项促销活动获得。如果没有为某个消费者提供促销活动，那么该消费者也不会购买该产品。从历史数据中可得出一个模型，它给出了任意一个消费者（特征向量 $\boldsymbol{x}$）对上述促销活动响应的概率

注 1：决策理论课程会将带你进入一系列有趣的相关问题。

注 2：在这里之所以使用目标市场营销，而非用户流失的例子，是因为我们还没有能力处理期望值框架在用户流失示例中所引出的复杂性问题。在第 11 章中，我们会做好准备，那时候再来讲解如何处理这个问题。

估计 $p_R(\boldsymbol{x})$。这个模型可以是分类树、逻辑回归或其他尚未谈及的模型。现在请考虑一下由特征向量 $\boldsymbol{x}$ 来描述的特定消费者是否会成为目标。

期望值的计算为分析提供了一个指导框架。具体来说，我们要计算面向消费者 $\boldsymbol{x}$ 进行目标市场营销的期望收益（或成本）：

$$\text{目标市场营销的期望收益} = p_R(\boldsymbol{x})\cdot v_R + [1 - p_R(\boldsymbol{x})]\cdot v_{NR}$$

其中，v_R 是消费者响应后我们获得的价值，v_{NR} 是消费者未响应时我们获得的价值。因为每个消费者要么响应，要么不响应，所以对其不响应的概率估计是 $1 - p_R(\boldsymbol{x})$。正如前面所提到的，这个概率值来自于历史数据，并体现在预测模型中。对收益 v_R 和 v_{NR} 的确认需要单独来进行，这是业务理解环节中的一部分（回顾第 2 章）。由于假设消费者只有响应促销活动才会购买产品，所以非目标市场营销目标消费者的期望收益是 0。

具体来说，假设消费者以 200 美元购买产品，而产品的相关成本为 100 美元。为了向消费者提供促销优惠，我们也需要支付一定的费用。假设我们邮寄了一些花哨的宣传材料，包括邮费在内的成本为 1 美元，如果消费者响应（购买产品），则产生的价值（利润）v_R 为 99 美元。如果消费者没有响应，那么 v_{NR} 的值会是多少呢？我们仍然邮寄了宣传材料，花费了 1 美元，相当于收益为 −1 美元。

现在，我们需要决定是否要向这个消费者提供促销优惠了：我们希望盈利吗？从技术上讲，精准广告的期望值（利润）是否大于 0？在数学上，它是这样表示的：

$$p_R(\boldsymbol{x})\cdot \$99 - [1 - p_R(\boldsymbol{x})]\cdot \$1 > 0$$

对公式稍作变换，就会得到一个决策规则：仅当消费者 $\boldsymbol{x}$ 满足以下条件时，对其提供特殊优惠。

$$p_R(\boldsymbol{x})\cdot \$99 > [1 - p_R(\boldsymbol{x})]\cdot \$1$$

$$p_R(\boldsymbol{x}) > 0.01$$

根据示例中的数值，只要估计的响应概率大于 1%，我们就应该把消费者认定为可能响应用户。

这体现了期望值如何指导我们使用模型，明确这一点有助于组织问题框架和对问题的分析。第 11 章将再次讨论这个问题。现在，本章将继续讨论期望值框架的另一个重要应用——分析这个基于数据建立的模型是否真的好用。

7.3.2 用期望值规范分类器的评估

此刻，我们希望把讨论的重点从个例决策转移到集体决策上。具体来说，我们需要评估模型在一系列情景下做出的一系列决策。为了在两个模型之间进行比较，这种评估是非常必要的。例如：该数据驱动模型是否比营销团队所建议的手工构建的模型性能更好？对于特定问题而言，分类树是否比线性判别模型更好？在解决诸如随机选择消费者作为目标市场营销目标的问题时，是否有哪个模型比基线“模型”更好？每个模型都有比其他模型做出更好决策的可能。我们关心的是：**总体来说**，每个模型的性能如何（它的期望值是多少）。

我们可以用刚刚介绍的期望值框架来确定每个模型的最佳决策，然后通过不同的方式用期

望值来比较模型。如果要计算组合模型的期望收益，那么公式 7-1 中的每个 o_i 都对应了一种不同的预测情况和实际情况的组合。我们希望汇总所有的可能情况：总体来说，当决定对消费者进行目标市场营销时，他们响应的概率是多少？他们不响应的概率又是多少？如果不对消费者进行目标市场营销，那么他们（假设被提供促销优惠时）会响应吗？你可能记得，我们其实已经在混淆矩阵中得出了计算上述问题所必需的数字。每个 o_i 都对应了混淆矩阵中的一个单元。例如：**预测为流失用户**同时又是**实际未流失用户**的组合概率是多少？这可以用测试集中落入矩阵单元 (Y, n) 的用户数量除以测试集中的用户总数来估计。

让我们在计算这些概率的过程中，从整个模型层面来计算期望收益。图 7-2 是模型归纳和模型评估过程中期望值计算的示意图。在图的左上方，训练集数据作为输入，进入归纳算法过程。在此基础上，我们建立起所要评估的模型，然后将该模型应用于所保留的测试集，并统计出混淆矩阵中不同单元所对应的计数的总和。表 7-4 展示了一个分类器混淆矩阵的具体案例。

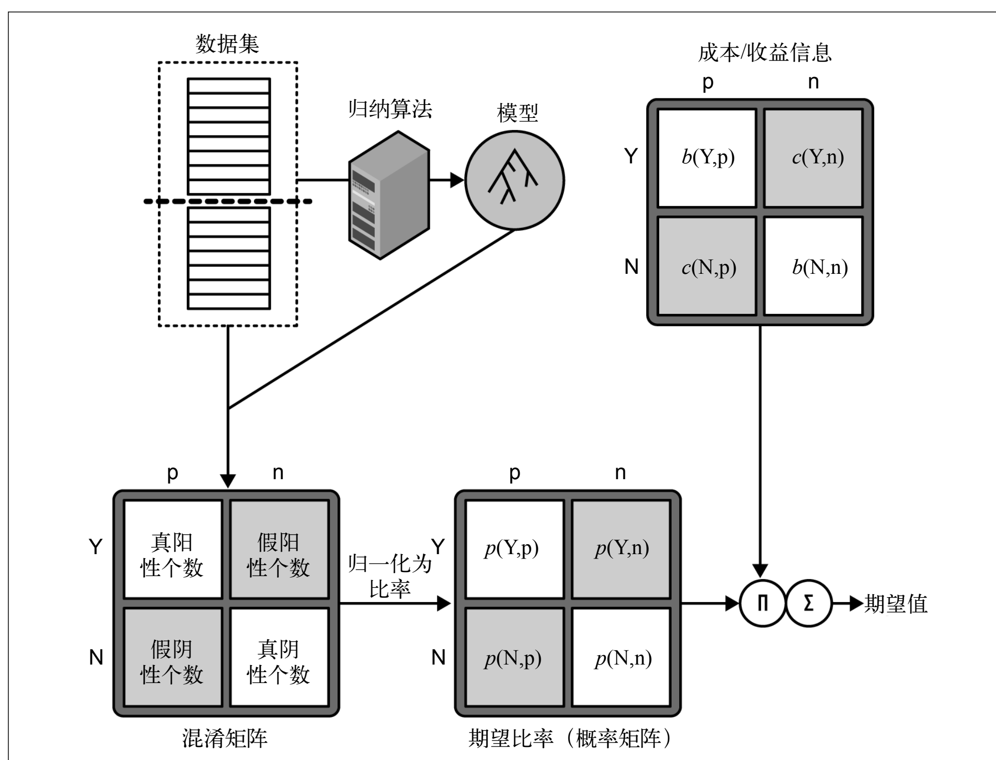


图 7-2：期望值计算图。Π 和 Σ 代表期望值计算中的乘法和求和

表7-4：一个混淆矩阵计数的案例

	p	n
Y	56	7
N	5	42

1. 错误率

在计算实际问题的期望值时，分析师经常面临这样的问题：这些概率来自于哪里？当你在测试集上验证模型的时候，答案就很明显了：这些（错误决策和正确决策的）概率可以通过在混淆矩阵中统计决策的正确率和错误率来估计。混淆矩阵的每个单元包含不同决策所对应的组合（预测的，实际的）的计数，我们将其表示为 count(h, a)（之所以使用 h 表示“预测的数目”，是因为 p 已经被使用了）。在期望值计算过程中，我们将这些计数转化为比率或估计概率 $p(h, a)$。我们通过用每个计数除以样本总数来进行转化：

$$p(h, a) = \text{count}(h, a) / T$$

下面是根据混淆矩阵中每个原始统计数据计算出的比率。这些比率就是我们将在公式 7-1 中计算期望值时使用的估计概率。

$$T = 110$$

$$p(\text{Y, p}) = 56/110 = 0.51 \quad p(\text{Y, n}) = 7/110 = 0.06$$

$$p(\text{N, p}) = 5/110 = 0.05 \quad p(\text{N, n}) = 42/110 = 0.38$$

2. 成本和收益

为了计算期望收益（见公式 7-1），我们还需要知道每对决策所对应的成本和收益的值。我们将构建一个与混淆矩阵维度相同（行和列）的成本收益矩阵[3]。成本收益矩阵详细列出了每对决策（预测，实际）对应的成本和收益（见图 7-3）。正确的分类（真阳性和真阴性）分别对应了收益 $b(\text{Y, p})$ 和 $b(\text{N, n})$；而错误的分类（假阳性和假阴性）分别对应“收益” $b(\text{Y, n})$ 和 $b(\text{N, p})$，而这实际上是成本（负收益），并且通常更明确地表示为成本 $c(\text{Y, n})$ 和 $c(\text{N, p})$。

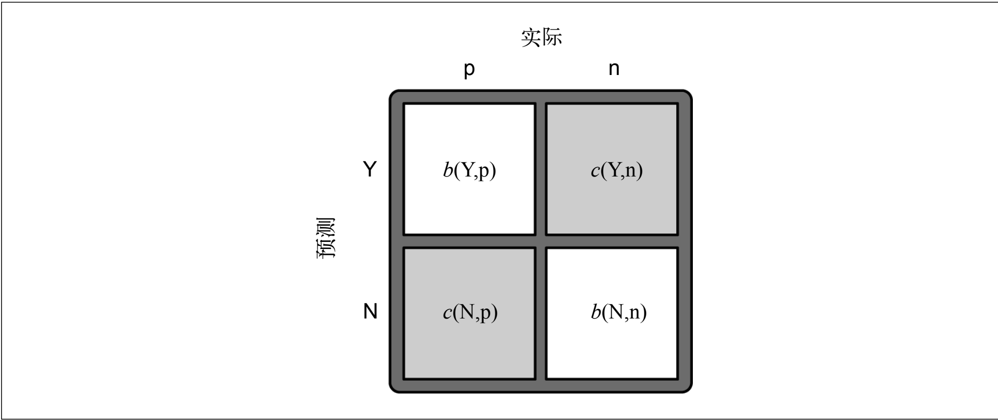
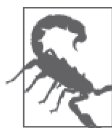


图 7-3：成本收益矩阵

通常，尽管我们可以从数据中估计概率，**却无法估计成本和收益**。我们会通过分析特定业务问题中决策导致的后果来确定成本和收益的值。实际上，确定成本和收益可能需要花费

注 3：有时候这个矩阵也被称为“代价矩阵”。——译者注

大量时间去思考。在许多情况下，我们并不能计算出一个准确的值，只能找到一个大概的范围。第 8 章将回过头来讨论在无法得出确切的值时该怎么做。比如说：对于用户流失问题而言，维护一个用户要花多少钱？这个价值可能取决于未来的手机使用情况，而不同用户之间的差异很大。用户之前的手机使用状况的数据也许有助于估计。在许多情况下，为了简化问题，我们会使用平均值而不是个别特定值来估计收益和成本。因此，下面的例子将忽略对个别用户特定成本 / 收益的计算，但是这个问题将在第 11 章中再讨论。



在数学上，除了符号之外，成本和收益之间是没有区别的。为了简单起见，从现在起，本章将所有值都表示为**收益**，成本就是负收益。这样，我们只需要定义一个函数，即 *b*(预测 , 实际)。

让我们回到目标市场营销的例子。这里的成本和收益是多少？简单起见，所有数字都将表示为美元。

- **假阳性错误**是指我们把一个用户认定为可能响应用户，并针对其进行营销，但其没有做出响应。我们已经说过，准备和邮寄宣传材料的成本是每人 1 美元。在这种情况下，收益为负值：$b(\mathrm{Y, n}) = -1$。
- **假阴性错误**是会购买产品的用户被错误判断为不会购买，因此没有对其进行产品宣传。这种情况下，我们既没有花费成本，也没有获得收益，因此 $b(\mathrm{N, p}) = 0$。
- **真阳性**是指给用户发放了宣传材料，该用户也购买了商品。这种情况下的收益是收入（200 美元）减去产品的相关成本（100 美元）和邮寄费用（1 美元），因此 $b(\mathrm{Y, p}) = 99$。
- **真阴性**的情况是未给没有购买产品意向的用户介绍产品。这种情况下收益是零（没有收入也没有成本），因此 $b(\mathrm{N, n}) = 0$。

这些成本收益的估计可归纳为一个 2 × 2 成本收益矩阵，如图 7-4 所示。请注意，这里的行和列与混淆矩阵是相同的，而这正是我们计算分类模型的总体期望值时所需要的。

$$\text{预测}\;\begin{matrix} \\ \mathrm{Y} \\ \mathrm{N} \end{matrix}\overset{\text{实际}}{\begin{pmatrix} \mathrm{p} & \mathrm{n} \\ 99 & -1 \\ 0 & 0 \end{pmatrix}}$$

图 7-4：目标市场营销示例的成本收益矩阵

给定成本和收益矩阵，将它与概率矩阵相乘，其结果之和就是总的期望利润。结果如下：

$$\begin{aligned}\text{期望收益} = {} & p(\mathrm{Y, p})\cdot b(\mathrm{Y, p}) + p(\mathrm{N, p})\cdot b(\mathrm{N, p}) + \\ & p(\mathrm{N, n})\cdot b(\mathrm{N, n}) + p(\mathrm{Y, n})\cdot b(\mathrm{Y, n})\end{aligned}$$

使用这个公式，就可以计算和比较各种模型或其他目标市场选择策略的期望收益。我们要做的只是用一组测试实例来计算混淆矩阵，并且以此计算出相应的成本收益矩阵。

这个公式用来比较分类器其实已经足够了，但是，我们还要沿着这条路继续前进，因为在实际应用中经常会用到一种替代方法。这个替代方法与一些使分类器性能可视化的技术密切相关（参见第 8 章）。另外，通过检查替代公式，我们可以清楚地知道如何处理本章开

头的模型的比较问题——一位分析师使用了有代表性（但不均衡的）数据集来测试模型性能，而另一位分析师使用了均衡数据集。

计算期望利润的一种常见方式就是分解出每个类别的概率，通常称为**类的先验概率**。类的先验概率，p(p) 和 p(n)，分别表示了出现正向结果和负向结果的可能性。把这些因素都考虑在内，我们可以把类别不平衡的影响与模型的基本预测能力区分开来。第 8 章将对此进行详细讨论。

基本概率规则是：

$$p(x, y) = p(y) \cdot p(x \mid y)$$

这表明两个事件同时发生的概率等于其中一个事件发生的概率乘以另一个事件在已知第一个事件发生的条件下发生的概率。使用此规则，我们可以重新计算期望收益：

$$\begin{aligned}\text{期望收益} = {} & p(\mathrm{Y|p}) \cdot p(\mathrm{p}) \cdot b(\mathrm{Y, p}) + p(\mathrm{N|p}) \cdot p(\mathrm{p}) \cdot b(\mathrm{N, p}) + \\ & p(\mathrm{N|n}) \cdot p(\mathrm{n}) \cdot b(\mathrm{N, n}) + p(\mathrm{Y|n}) \cdot p(\mathrm{n}) \cdot b(\mathrm{Y, n})\end{aligned}$$

考虑到类先验 p(p) 和 p(n)，我们得到最终的公式。

公式 7-2：*考虑了先验 p(p) 和 p(n) 的期望收益公式*

$$\begin{aligned}\text{期望收益} = {} & p(\mathrm{p}) \cdot [p(\mathrm{Y|p}) \cdot b(\mathrm{Y, p}) + p(\mathrm{N|p}) \cdot b(\mathrm{N, p})] + \\ & p(\mathrm{n}) \cdot [p(\mathrm{N|n}) \cdot b(\mathrm{N, n}) + p(\mathrm{Y|n}) \cdot b(\mathrm{Y, n})]\end{aligned}$$

在这个繁杂的公式中，我们注意到有一部分（第一部分）对应了正向实例的期望收益，另一部分（第二部分）则对应了负向实例的期望收益。各部分的所加权重为该种实例出现的概率。因此，如果正样本非常少，那么它们对总体期望收益的相应贡献就会很小。在这个替代公式中，p(Y|p) 对应真阳性比率，p(Y|n) 对应假阳性比率，以此类推。这个概率可以直接从混淆矩阵计算（请参阅后文的"其他评估指标"）。

表 7-5 展示了我们的混淆矩阵。

表7-5：我们的混淆矩阵（原始计数）

	p	n
Y	56	7
N	5	42

表 7-6 显示了我们需要的类的先验概率和各种错误率。

表7-6：类的先验概率和真阳性比率、假阳性比率等

$T = 110$	
$P = 61$	$N = 49$
p(p) = 0.55	p(n) = 0.45
真阳性比率 = 56/61 = 0.92	假阳性比率 = 7/49 = 0.14
假阴性比率 = 5/61 = 0.08	真阴性比率 = 42/49 = 0.86

让我们回到目标市场营销的示例。模型计算出的期望利润是多少？可以用公式 7-2 计算：

$$\begin{aligned}
\text{期望收益} &= p(\mathrm{p})\cdot[p(\mathrm{Y}|\mathrm{p})\cdot b(\mathrm{Y,p}) + p(\mathrm{N}|\mathrm{p})\cdot b(\mathrm{N,p})] + \\
&\quad p(\mathrm{n})\cdot[p(\mathrm{N}|\mathrm{n})\cdot b(\mathrm{N,n}) + p(\mathrm{Y}|\mathrm{n})\cdot b(\mathrm{Y,n})] \\
&= 0.55\cdot[0.92\cdot b(\mathrm{Y,p}) + 0.08\cdot b(\mathrm{N,p})] + \\
&\quad 0.45\cdot[0.86\cdot b(\mathrm{N,n}) + 0.14\cdot b(\mathrm{Y,n})] \\
&= 0.55\cdot[0.92\cdot 99 + 0.08\cdot 0] + \\
&\quad 0.45\cdot[0.86\cdot 0 + 0.14\cdot(-1)] \\
&= 50.1 - 0.063 \\
&\approx \$\mathbf{50.04}
\end{aligned}$$

这个期望值意味着，如果将这个模型应用于潜在客户群，并给那些被归为正向标签的用户邮寄宣传材料，那么我们预期能从每个用户身上赚到 50 美元。

现在，我们学会了一种能够处理本章开头所提到的激励问题的方法：计算模型的期望值，而不是其准确率。另外，使用这种替代方法，即便一个分析师使用了代表性数据集，而另一个使用平衡数据集来进行测试，我们仍可以比较两个模型。在每次计算中，我们都可以简单地替换先验概率，使用平衡的数据分布来对应 $p(\mathrm{p}) = 0.5$ 和 $p(\mathrm{n}) = 0.5$ 的概率分布。精通数学的读者可以尝试证明，即便先验测试集发生变化，公式中的其他因素也不会改变。



为了结束有关估计利润的这部分内容，在此强调一下在计算成本收益矩阵时常见的两个陷阱。

- 保持成本收益矩阵中符号的一致性非常重要。本书将收益看作正向的，而把成本看作负向的。而在许多数据挖掘的研究中，最重要的往往不是使利润最大化，而是使成本最小化，因此成本和收益的符号是相反的。虽然在数学上这没有区别，但是，有一个统一的视角是非常重要的。
- 计算成本收益矩阵时容易犯的一个错误，就是“重复计算”，增加收益的同时也减少了成本（反之亦然），而比较有效的检验方式是计算因为决策改进而带来的**收益提升**。

  假设你已经建立了一个模型来预测哪些账户遭到了欺诈。你设定一项欺诈事件的平均成本为 1000 美元。若检测出欺诈的收益为每项 +1000 美元，而未能检测出欺诈的成本是 −1000 美元，那么每检测出一项欺诈的**收益提升**是多少呢？ 你会这样计算：

  $b(\mathrm{Y, p}) - b(\mathrm{N, p}) = \$1000 - (-\$1000) = \$2000$

  但在直觉上，你知道实际带来的增长只有 1000 美元，因此这表明你重复计算了。解决方案是，规定捕获欺诈的收益是 +1000 美元，或捕获丢失欺诈的成本是 −1000 美元，两者不能同时计入，其一应该是零。

其他评估指标

你可能会在学习数据科学的过程中遇到许多评估指标。其实，所有的这些指标都是在混淆矩阵的基础上建立起来的。参考混淆矩阵中每个单元的含义，我们分别用 *TP*、*FP*、*TN* 和 *FN* 来表示真阳性、假阳性、真阴性和假阴性，然后用这些单元来计算各种评估指标。**真阳性比率**和**假阴性比率**分别指当被预测的个体真实值为正的时候，预测值为正确（预测为正）和错误（预测为负）的比率，表示为 $TP/(TP+FN)$ 和 $FN/(TP+FN)$。**真阴性比率**和**假阳性比率**是在被预测的个体真实值为负的时候相应的比率。这些通常被看作当个体真实值为 p 的时候，预测为 Y 的概率估计，即 $p(\mathrm{Y}|\mathrm{p})$，诸如此类。我们将在第 8 章中继续探讨这些测量方法。

经常使用的指标还有精确度和召回率，它们在文本分类和信息检索的场景中尤其常用。召回率与真阳性率相同，而精确度则是 $TP/(TP+FP)$，即预测为正的情况下的准确率。F-measure 则是某给定点的精度和召回的调和平均值：

$$\text{F-measure} = 2 \cdot \frac{\text{精确度} \cdot \text{召回率}}{\text{精确度} + \text{召回率}}$$

统计学、模式识别和流行病学等许多领域的从业者都会用到分类器的敏感性和特异性：

$$\text{敏感性} = TN/(TN+FP) = \text{真阴性比率} = 1 - \text{假阳性比率}$$
$$\text{特异性} = TP/(TP+FN) = \text{真阳性比率}$$

你可能还听过**阳性预测值**，这其实和精确度相同。

前面提到的准确率，则是预测正确的计数除以样本总数，或表示为：

$$\text{准确率} = \frac{TP+TN}{P+N}$$

Swets（1996）列出了许多其他评估指标以及它们与混淆矩阵的关系。

7.4 评估、基线性能以及对数据投资的意义

到目前为止，我们已经对模型的评估进行了相对孤立的讨论。在某些情况下，仅仅证明模型可以产生**一些**（非零）利润，或者投资获得了正向的收益的过程本身就富含信息。然而，这里需要提出数据科学中的另一条基本概念：**仔细考虑什么才是合适的模型性能的比较基线是很重要的。**这对于数据科学团队来说非常重要，因为他们要了解模型性能是否确实有所提高。这对向利益相关者展示挖掘数据的附加价值也同样重要。那么，什么才算是合适的比较基线？

答案当然取决于实际应用。提出合适的基线是数据挖掘流程中业务理解环节的一项重要任务。不过，仍然有一些通用的原则可供参考。

对于分类模型而言，模拟完全随机的情况并以此测量模型性能是非常容易的。第 8 章将讨

论的随机分类框架有自然基线，可以展示随机分类所要达到的指标，这对一些非常困难的问题或初步探讨会非常有用。与随机模型的比较往往会证明数据中仍有待提取的信息。

然而，因为击败随机模型可能很容易（或者看起来很容易），所以证明随机模型的优越性可能不是一件有趣的事情，也无法带来什么信息。因此数据科学家通常需要用替代模型，通常是简洁且不过度简单的模型，以便验证继续数据挖掘工作的合理性。

Nate Silver 在他的作品《信号与噪声》（2012）中，提到了天气预报的基线问题：

> 任何天气预报要想证明其价值，必须通过两个基本测试：首先，它必须要比气象学家所说的持续性（即假设明天和后天的天气与今天一样）更准确；其次，它还必须要击败气候学理论，即特定地区特定日期的长期历史平均条件。

换句话说，天气预报员有两个可以用于比较的简洁且不过度简单的基线模型：一个（持续）预测明天的天气将会和今天一样，另一个（气候学）预测这一天的天气就是往年的平均历史天气。这两个模型的性能都比随机预测要好得多，而且也都很容易计算，因此可以作为用于比较的自然基线。任何更复杂的新模型都必须击败这两个模型。

好的基线的一般性原则是什么？对于分类任务而言，一条好的基线必须是一个**大样本分类器**，即一个总是选择训练数据集中的多数类的原始分类器（参见 5.3.1 节中的基础比率注释）。这部分内容看上去可能十分浅显，可以略过，但是它其实值得我们花一点时间阅读，因为很多非常聪明又有分析性思维的人常因为忽略了这个地方而遇到麻烦。例如，分析师看到一个分类器的分类准确率为 94%，就认为模型的性能很好——但实际上正样本只有 6%。因此，一个简单的大样本分类器也会有 94%的准确率。实际上，很多刚开始研究数据科学的学生都会惊讶地发现，他们根据数据构建的模型只是简单地把一切都预测为数据集中的多数类。值得注意的是，如果建模过程以将模型的准确率最大化为目标，那么这种现象可能是有意义的——模型的准确率很难超过 94%。这里要运用本章的核心思想：仔细考虑我们想从数据挖掘的结果中获得什么。追求预测准确率的最大化通常不是一个合适的目标。如果这是算法现在正在做的，那我们可能使用了错误的算法。针对回归问题有一条类似的基线：使用总体的均值（通常是平均值或中位数）作为预测值。

在一些应用场景中，我们可能需要组合多个简单平均值。例如，在评估那个用于预测特定用户将为特定电影打多少颗“星”的推荐系统时，我们可以获得一部电影在整个总体中获得的平均星数（观众的喜好程度）和特定用户给出的平均星数（该用户的整体偏见是什么）。基于这两者的预测要比基于其中单独一个的预测好得多。

除了这些简单的基线模型之外，稍微复杂的替代方法是仅考虑非常少的特征信息的模型。例如，回顾从第 3 章开始介绍的第一个数据挖掘示例：寻找富信息变量。如果我们找到一个与目标有最佳关联的变量，就可以建立一个只有该变量的分类或回归模型，这给出了基线性能的另一种观点：简单的“有条件”模型性能如何？这里的“有条件”意味着基于特征值或以特征值为条件进行不同的预测。因此，总体的平均值有时被称为“无条件”的平均值。

从数据中挖掘这类单特征预测模型的一个例子就是用树型归纳构建“决策树桩”——仅有一个内部节点（根节点）的决策树。只有一个内部节点的树意味着在树的归纳过程中，会选择信息量最大的特征来做决策。Robert Holte（1993）在他的一篇著名的机器学习论文中

表示，在机器学习的研究中，决策树桩在许多测试集上都会表现出相当好的基线性能。决策树桩是一个从很多可用信息中选择最有效的信息（见第 3 章）的策略的例子，而所有的决策也都根据这个策略来进行。在某些情况下，大部分的影响可能来自某单个特征，而且这个方法可以评估该特征是否造成影响以及造成多大影响。

这个概念可以扩展到数据**源**当中，而且与第 1 章中提到的基本原则（我们应该把数据看作要投资的资产）相关。如果你正在考虑使用各个来源收集到的数据去构建模型，就应该把这个结果与基于单独来源的数据建立的模型做比较。通常，你需要大量成本来获取新的数据源。某些情况下，它们是实际的金钱成本；另一些情况下，这还关系到管理与数据供应商关系和监督数据馈送的人员的时间成本。因此，针对每个数据源，数据科学团队应该在使用这个数据源的模型与不使用这个数据源的模型之间进行比较。通过比较，我们可以量化每个数据源所提供的价值以估计成本。如果某个数据源所带来的价值可以忽略不计，那么团队可以舍弃它，从而降低成本。

除了与简单模型（和简化数据模型）比较之外，基于行业知识或“已知经验”来构建简单且低成本的模型以供比较也是非常有用的。例如，在一个欺诈检测的应用场景中，大多数被欺诈的账户通常会出现交易量突然增加的情况，因此通过检查账户的交易数量和交易额是否突然增加，我们可以捕获大部分的欺诈事件。这个想法很容易实现（这本质上是一个单变量预测模型），而且它提供了一条有用的比较基线，可以充分证明数据挖掘的优势。类似地，IBM 的团队经常利用数据挖掘来指导他们的销售工作，他们部署了一个简单的销售模型：根据其之前的收入对现存客户进行排序，而根据年销售额对其他公司进行排序。[4] 他们可以证明其所执行的数据挖掘带来的价值超过这项简单策略带来的价值。无论数据挖掘小组选择了什么样的比较基线，它都应该能让利益相关者觉得其中的信息很有用，而且最好很有说服力。

7.5 小结

数据科学一个至关重要的环节就是对模型进行正确的评估。但是令人惊讶的是，特别正确的模型评估是很难实现的，而评估过程通常需要进行多次迭代。人们往往倾向于选择简单的评估指标，比如分类准确率，因为它们不仅很容易计算，又在许多研究论文中被使用，还可能是人们在学校学到的东西。然而在现实中，过于简单的方法很少能够捕捉到问题真正的关键，甚至常常误导我们。相反，数据科学家应该仔细思考模型将会如何应用于实践，并且设计出合适的度量指标。

期望值的计算过程为组织这种思路提供了良好的框架。它将有助于构建评估框架，并且一旦最终部署的模型产生了不可接受的结果，它也有助于识别错误。

在评估数据科学结果时，必须仔细考虑数据的特点。例如，真正的分类问题通常会出现非常不平衡的类别分布（也就是说，类别不会普遍地等比例出现）。调整类别的比例对训练模型可能是有用的（甚至是必要的），但是，评估的时候还是应该使用原始、真实的数据集，以便结果能够反映出真正要实现的目标。

注 4：他们将这些称为 Willy Sutton 模型，这位著名的银行抢劫犯抢劫了银行，因为“这就是放钱的地方”。

要计算模型的总体期望值，必须明确决策的成本和收益。如果能做到这一点，那么数据科学家就可以计算出每个模型中的每个实例的期望成本，并选择期望成本最低或利润最大的模型。

同样至关重要的一个问题是：我们应把数据驱动模型与什么做比较，来判断它是否性能良好或者是否性能更好。这个问题的答案与对业务的理解紧密相连，不过，仍然有各种最佳实践需要数据科学团队遵循。

我们应用前几章所展示的概念阐释了本章的思想。这些概念当然是一般性的，而且与最初的基本概念相关：数据应被视为资产，我们应该思考如何对其进行投资。这一点可以体现在本章的简要讨论中：分析人员不仅可以在不同的模型和不同的基线之间进行比较，还可以比较不同的数据源所产生的结果。不同的数据源会有不同的相关成本，而谨慎的评估可以告诉我们选择哪一个会使投资回报最大化。

最后，本章讨论了衡量模型性能的单一数值指标。它们可以回答诸如“我期望有多少收益”“我应该使用模型 A 还是模型 B”之类的问题。相应的答案虽然有用，但是仅能提供基于一系列具体假设的“单点数值”。如果能够在更宽泛的条件下，将模型的行为可视化，则往往会更有启发性。下一章就将讨论这一点：模型性能的图形化。

第8章

模型性能的可视化

基本概念：多种不确定性下的模型性能的可视化；进一步考虑期望从数据挖掘的结果中获得什么

示例方法：利润曲线；累积响应曲线；提升曲线；ROC 曲线

上一章介绍了模型评估的基本问题，并探讨了如何建立一个好模型，还基于期望值的框架实现了进一步的计算。比起之前的章节，上一章的数学味道更浓一些，因此如果你是第一次学习那些知识，那么可能很难理解其中的公式。虽然这些公式是后续章节的基础，但它们本身可能不够直观。本章将从另一个角度来进一步理解这些公式。

期望利润公式（见公式 7-2）需要一系列特定条件来得出该场景下由单个数字表示的期望利润值。但数据科学团队之外的利益相关者可能缺乏耐心，不想考虑这些细节，只想看到关于模型性能的更高层次的、更直观的描述。因为这些指标依赖于严格的假设条件（比如成本和收益的确切信息，或者准确的模型概率估计），所以就连擅长和公式及枯燥的计算打交道的数据科学家也会觉得这样的单一检验干巴巴的，信息量太小。一般来说，可视化形式往往是比数学计算形式更有效的呈现方式，本章会介绍一些有用的技术。

8.1 排序，而不是分类

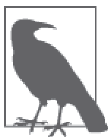
7.3 节讨论了如何基于各种情形的期望值，利用模型分配的分数来为每个情形做决策。而另一种决策策略是按分数对不同情形**排序**，然后按照业务逻辑对排序靠前的几种情形采取适当的措施。我们不会单独判断每种情形，而是选择前 *n* 种情形（或选择所有分数大于给定阈值的情形）。实践中，这样做的原因有很多。

原因之一是模型给出的分数虽然能够根据每种情形属于某类的可能性的大小，对各种情形进行排序，却并不是真实概率（回顾第 4 章所讨论的，把到分类边界的距离作为分类器的

分数）。值得强调的一点是，某些原因可能导致我们无法通过分类器得到准确的概率估计。比如在目标市场营销中，当获取不到足够的有代表性训练实例时，这种情况就会发生。虽然概率估计并不与响应概率完全对应，但分类器的分数仍非常有助于判断哪种情形更好。

一个常见场景是，如果你有一些活动**预算**，比如针对某个活动的固定营销预算，那么你一定想针对那些最有希望响应的用户进行营销。如果要根据（不随类别变化的）成本和收益对期望值最高的用户投放营销广告，那么你只需要按照可能性的高低对目标类别进行排序，而无须追求精确的概率估计。唯一需要注意的是，你的预算应该尽量小，以免得到负的期望值。目前，我们还是把它看作业务理解业务。

还有一个原因是，虽然成本和收益通常无法被精确地定义，但是不管怎样我们还是愿意采取行动（尤其是针对那些最有可能响应的用户）。下一节将继续讨论这个问题。



如果**单个**情况的成本和收益各不相同，那么 7.3 节对期望值的讨论就应该说明仅按照可能性排序是不够的。

在为实例评分时，在某些情况下，分类器应当保守地进行决策，因为其在预测时需要有非常大的把握。这相当于给输出分数设置了很高的阈值。相反，阈值越低，分类器的自由度越大。[1]

这其实把问题复杂化了，为此我们需要对原来用于评估和比较模型的框架进行拓展。7.1.2 节提到，分类器会产生混淆矩阵。一个**带有阈值**的排序分类器对应一个混淆矩阵。混淆矩阵会随着阈值的改变而改变，因为真阳性和假阳性的数量发生了变化。

图 8-1 阐述了这个基本思想。随着阈值的降低，实例逐渐从混淆矩阵中的 N 行上升到 Y 行：原来被判定为负向的实例现在被认定为正向，实例数目也因此改变。至于哪种实例数目发生改变，取决于实例到底属于哪个类别。如果实例本身是阳性（p 列），那么就会上升到真阳性 (Y, p) 的单元；如果为阴性（n 列），则会上升到假阳性 (Y, n) 单元中。严格来讲，不同的阈值会产生不同的分类器，对应不同的混淆矩阵。

注 1：的确，在某些应用场景中，同一模型的评分可以通过改变阈值在不同情况下产生不同决策。比如，同一个模型既可以用于批准或拒绝信贷发放，也可以用于给新用户的授信。

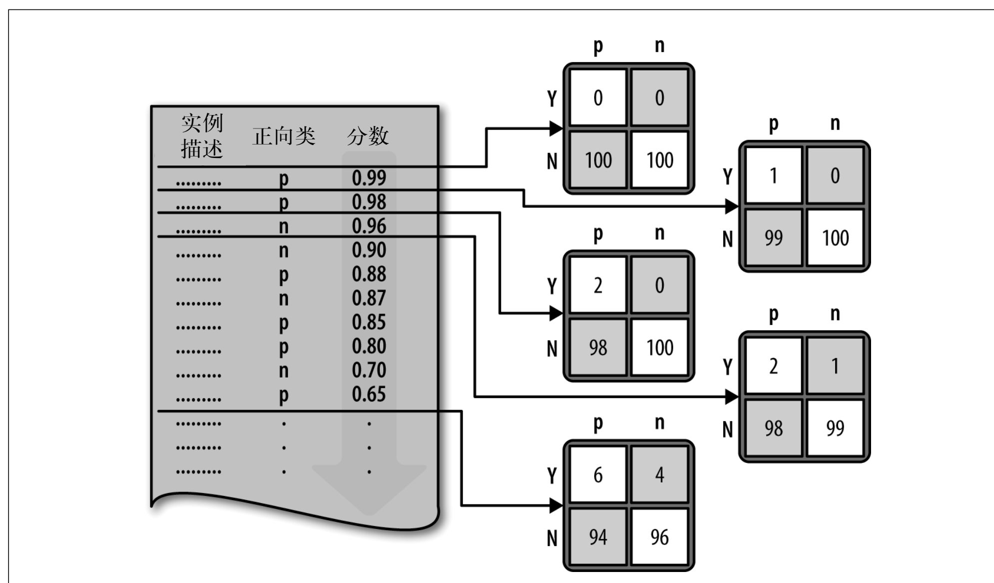


图 8-1：给按分数排序的不同实例设定阈值。这里有一些被模型赋予分数并按照分数降序排列的测试实例，我们设定一系列阈值（即每条水平线），使大于阈值的实例为正，小于阈值的实例为负。每个阈值将产生一个特定的混淆矩阵

这里有两个问题。如何对不同的**排序**进行比较？如何选择合适的**阈值**？如果有准确的概率估计和定义明确的成本收益矩阵，那么在讨论期望值的过程中，我们就已经回答了第二个问题：我们会在期望收益超过某个期望水平（通常是 0）时设定阈值。让我们来仔细探讨并扩展这个概念。

8.2 利润曲线

我们从 7.3 节中知道了如何计算期望利润，刚刚又学习了如何用模型来对实例进行排序。把以上两种思想相结合，就可以以曲线的形式构建出各种体现模型性能的可视化图像。每条曲线的绘制都基于这样一种效果检验：如果按顺序把一系列数据点设为分类器的分类阈值，那么分类器会把数据点以不同的方式分类（正或负）。在按照顺序逐渐降低阈值的过程中，被预测为正的实例会越来越多，而被预测为负的实例会越来越少。每个阈值（即每组正实例和负实例）都对应一个混淆矩阵。从前面的章节可以知道，如果我们有一个混淆矩阵，也知道对应决策的成本和收益信息，就能得出该矩阵对应的期望值。

更确切地说，只要有排序分类器，我们就能得到许多实例的预测分数，并按照分数将其降序排序，然后测量每个基于所选连续分割点得出的期望收益值。概念上，这相当于按分数将列表中的实例降序排序，并且按从上到下的顺序在每一个实例后记录其对应的期望利润。在每一个分割点处，都要记录列表中被预测为正的实例的比例和对应的收益估计值。将这些值绘制成图像，就得到了**利润曲线**。图 8-2 展示了三条这样的利润曲线。

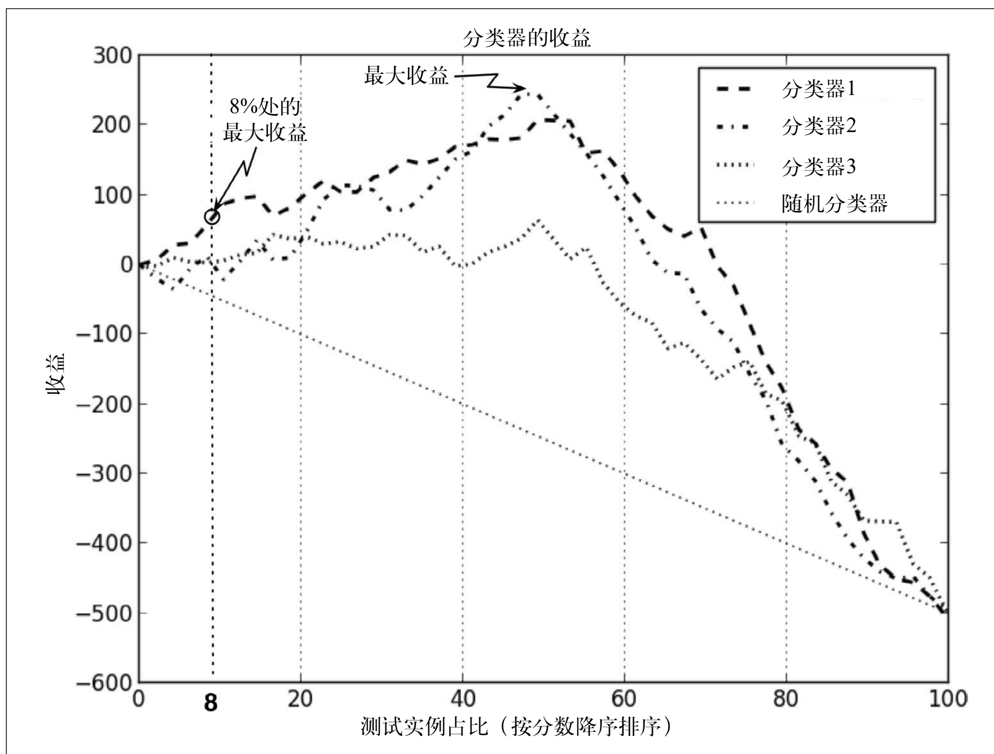


图 8-2：三种分类器的利润曲线。每条曲线表示随着目标用户在用户总体中的比例增大，期望累积利润的变化

图 8-2 是基于包含 1000 名用户的测试数据集而构建和绘制的，该测试集也可以被认为是对之前进行过试销售的人群的一小部分随机抽样。（在解释结果时，我们通常会关注用户占整体的比例，以便将结果推广到总体。）在每条曲线上，我们根据某个模型按照接受某个优惠活动的概率由高到低对用户排序。在这个示例中，假设边际利润很小——每个优惠名额预留和推广的成本是 5 美元，用户接受优惠后的收益是 9 美元，利润为 4 美元。则其对应的成本收益矩阵为：

	p	n
Y	$4	−$5
N	$0	$0

从曲线中可以看到，负利润有时（不是一直）还会出现负利润，这取决于成本以及类别比率。负利润尤其会出现在利润率较低、回应者较少的时候。当把阈值设定得过低时，模型会对过多不会响应的用户发出促销优惠，从而导致成本过高 [2]，因此曲线会显示出“赤字”。

你可能会注意到，四条曲线的起始点和终点都是相同的。这一点应该很容易理解，因为最

注 2：为简化问题，我们将忽略库存及其他现实因素，否则利润的计算会变得复杂。

左侧没有用户被视为目标，所以没有支出和收入；而最右侧，因为所有的用户都是目标，所以所有分类器的结果都相同。两点之间的差别则取决于分类器对用户的排序情况。其中随机分类器性能最差，因为它选择响应者和未响应者的概率相同。这些被测试的分类器中，分类器 2 在对前 50% 的用户提供优惠时，能够获取最高 200 美元的利润。那么，如果你仅以利润最大化为目标，并且有充足的资源可利用，你就应该用这个分类器给用户打分，然后把前 50%（最高的 50%）的用户作为优惠对象。

接下来，考虑一个稍有不同但十分常见的场景——**预算受限**。你手头的可用资金是固定的，在盈利之前，你必须慎重考虑如何支配这些资金。这种情况在营销活动中很常见。就像前面所说的那样，你还是想针对那些排序较靠前的用户进行营销，但现在你的预算有限[3]，这可能会影响你的策略。假设你一共有 10 万名用户，营销活动的预算为 4 万美元，你要用建模的结果（见图 8-2 的利润曲线）来找出分配预算的最佳方式，那么该怎么做呢？首先，你要计算出你能发出多少份优惠。如果每份优惠花费 5 美元，那么你最多能针对 $40 000/$5 = 8000 名客户进行营销。虽然你仍旧想要找出最有可能响应的客户，但是不同模型对客户的排序也不相同，这次营销活动该用哪一个模型呢？ 8000 名客户是整个客户群的 8%，因此你要在性能曲线中找到 x = 8% 的位置。在这个点上，性能最好的模型是分类器 1，因此你该选择这个模型来对整个总体打分，然后对排序靠前的前 8000 名客户发放优惠。

综上，通过这个示例，我们知道了，预算限制不仅会改变操作点（从总体实例的 50% 变到 8%），还会改变对排序分类器的选择。

8.3 ROC图像和曲线

利润曲线仅在你对所使用的分类器的假设条件很确定的前提下才适用。而收益的计算中有两个特别值得注意的关键条件。

(1) **类的先验概率**，就是目标群体中正实例和负实例的比例，有时也被称为**基础比率**（通常指正实例的比例）。回想一下公式 7-2，它对 p(p) 和 p(n) 很敏感。

(2) **成本和收益**。期望利润对成本收益矩阵中各单元的成本和收益的相对水平尤其敏感。

如果类的先验概率和成本收益的估计值都是已知且稳定的，那么利润曲线对模型性能可视化来说可能是一个不错的选择。

然而在很多领域中，这些条件都是不确定或不稳定的。比如在欺诈检测中，欺诈的数量随时间和地点的改变而改变（Leigh, 1995; Fawcett & Provost, 1997），这种改变会影响先验概率。而在手机用户流失管理示例中，营销活动的预算和提供优惠的成本不同，预期成本也会不同。

处理不确定因素的一种方法是，让每个模型生成很多不同的预期利润值。不过这种方法有些不尽如人意，因为当模型、类的先验概率和决策成本组合起来时问题的复杂程度也会加

注 3：另一种常见情况是劳动力受限，与预算受限相似，因为你用于解决问题的可分配资源（财力或人力）有限，所以你想“花最少的资源，办最大的事”。比如说，因为你手下的欺诈分析师有限，所以你想让他们处理可能性最大的疑似欺诈案件。

倍。对于分析师来说，他们很难在短时间内处理大量的利润曲线，理解各种含义，并对利益相关者进行解释。

还有一种处理不确定性的方法，就是展示整个模型性能概率的空间，比如受试者工作特征（下称 ROC）图像（Swets, 1988; Swets, Dawes & Monahan, 2000; Fawcett, 2006）。ROC 图像是分类器的二维图像，*x* 轴为假阳性比率，*y* 轴为真阳性比率，描绘的是分类器在收益（真阳性）与成本（假阳性）之间的权衡。图 8-3 的 ROC 图像包含从 A 到 E 5 个分类器。

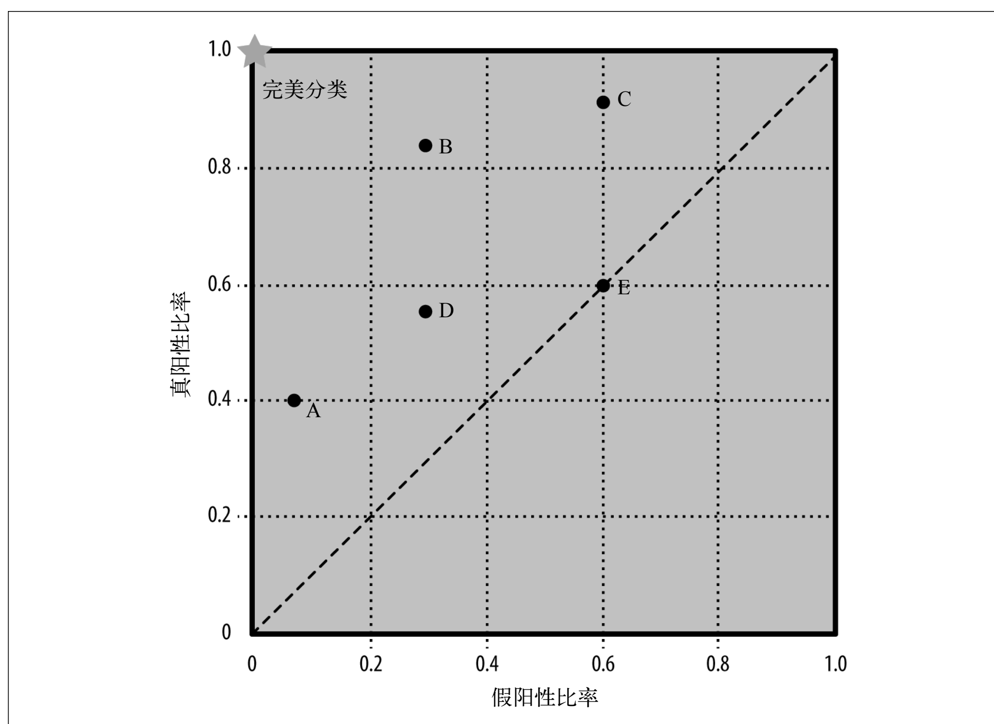


图 8-3：ROC 空间，5 个分类器（A~E）的性能以坐标点形式绘制在图中

离散分类器是一个只会输出类标签（而不是排序）的分类器。正如前面提到的，每个这样的分类器都会生成混淆矩阵（由真阳性、假阳性、真阴性、假阴性的数字和比率组成）。值得注意的是，虽然混淆矩阵包含了四个数字，但是我们其实只需要两个比率：真阳性比率和假阴性比率中的一个，以及假阳性比率和真阴性比率中的一个。因为每组两者和为 1，所以只要给定其中一个，就能求出另一个。通常我们会选择真阳性比率（tp rate）和假阳性比率（fp rate）。为了保证 ROC 曲线的合理性，我们在绘制时也会选择这两个比率。每个离散分类器都会产生一个 (**真阳性比率** , **假阳性比率**) 的组合来对应 ROC 空间中的一个点。图 8-3 中的分类器都是离散分类器。这里值得说明一下，**真阳性比率**的计算只需用到实际为正的实例，而**假阳性比率**的计算只需用到实际为负的实例。



虽然记清楚**真阳性比率**和**假阳性比率**所对应的统计量，对那些不常处理这些问题的人来说可能会比较困难，但是记忆不那么正规又很直观的称谓就会简单得多。**真阳性比率**有时也叫**命中率**，即分类器辨别正确的、实际为正的实例的比例。**假阳性比率**有时也叫**误警率**，即分类器辨别错误的、实际为负（即预测为正）的实例的比例。

ROC 空间中有几个坐标点需要注意：左下方的 (0, 0) 代表从未预测到正实例的策略，这样的分类器不会犯假阳性的错误，但也不会出现真阳性的情况。与之相反的是无条件预测为正的策略，即右上角的 (1, 1)。而 (0, 1) 代表完美分类，在这里用星号表示。将 (0, 0) 与 (1, 1) 连接起来的对角线代表预测类别的方法。比如，如果某分类器在一半情况下会随机将实例预测为正，那么就可以认为该分类器能正确地预测一半正实例和一半负实例，对应 ROC 空间中的 (0.5, 0.5)；如果分类器在 90% 的情况下将实例预测为正，那么就可以认为该分类器能正确预测 90% 的正实例，但是假阳性率也会上升至 90%，对应 ROC 空间中的 (0.9, 0.9)。因此，随机分类器在 ROC 空间中对应的点会在对角线上来回移动，其位置取决于分类器将实例预测为正的比率。为了使点从对角线转移到左上三角区域，分类器必须从数据中发掘出一些信息。图 8-3 中，点 (0.6, 0.6) 处的分类器 E 几乎是随机分类的，可以说它在 60% 的情况下都会将实例预测为正。注意，分类器对应的点不应处于 ROC 图像的右下三角区域中，因为这意味着分类器的预测效果比随机预测还要差。

在 ROC 空间中，如果一个点在另一个点的左上方（前者的**真阳性比率**高于后者，同时**假阳性比率**不低于后者；或前者的**假阳性比率**低于后者，同时**真阳性比率**不高于后者，或两种都更好），则前者优于后者。ROC 图像左边、接近 *x* 轴的分类器较为“保守”，因为它们仅在有足够多的证据时才会报警（做正分类），所以它们很少犯假阳性错误，但真阳性率也较低；图像右上方的分类器较为“自由”，因为它们将实例预测为正的门槛较低，所以它们几乎能把所有的正实例预测正确，但同时假阳性率也会较高。由此，在图 8-3 中，A 比 B 保守，而 B 又比 C 保守。因为许多现实领域中都有大量的负实例（见 7.1 中“坏的正实例与无害的负实例”中的讨论），所以图中最左侧的分类器比其他的更为有趣。在负实例较多的情况下，即使分类器误警率适中，情况也有可能失去控制。排序模型会在 ROC 图像中生成一系列点（连成一条曲线）。前文提到，设置阈值后的排序模型可以生成离散（二元）分类器：如果分类器的输出结果超过该阈值，那么结果为 Y，否则为 N。每个阈值都对应 ROC 空间中的一个点，如图 8-4 所示。

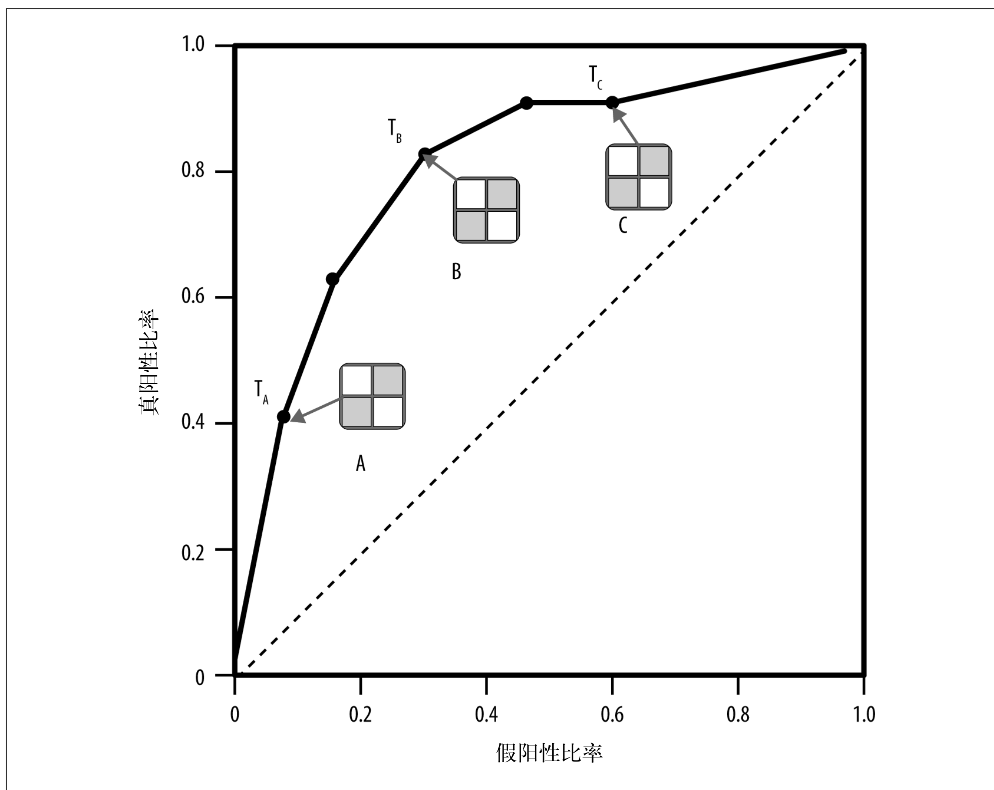


图 8-4：ROC 空间中的每个点都对应一个混淆矩阵

从概念上讲，我们可以按分数对实例进行分类，并改变阈值（从 –∞ 到 +∞），同时跟踪曲线在 ROC 空间中的移动。如图 8-5 所示。在遇到正实例的时候点往上移动（增加真阳性），遇到负实例时往右移动（增加假阳性），这条“曲线”实际上就是单个测试集的阶梯函数。如果数据量足够大，曲线就会较为平滑。[4]

注 4：从技术上讲，如果在一次运行中有许多实例分数相同，我们就应该计算一下整个运行过程中正实例和负实例的个数，从以使 OC 曲线呈平滑状，而不是方形阶梯状。

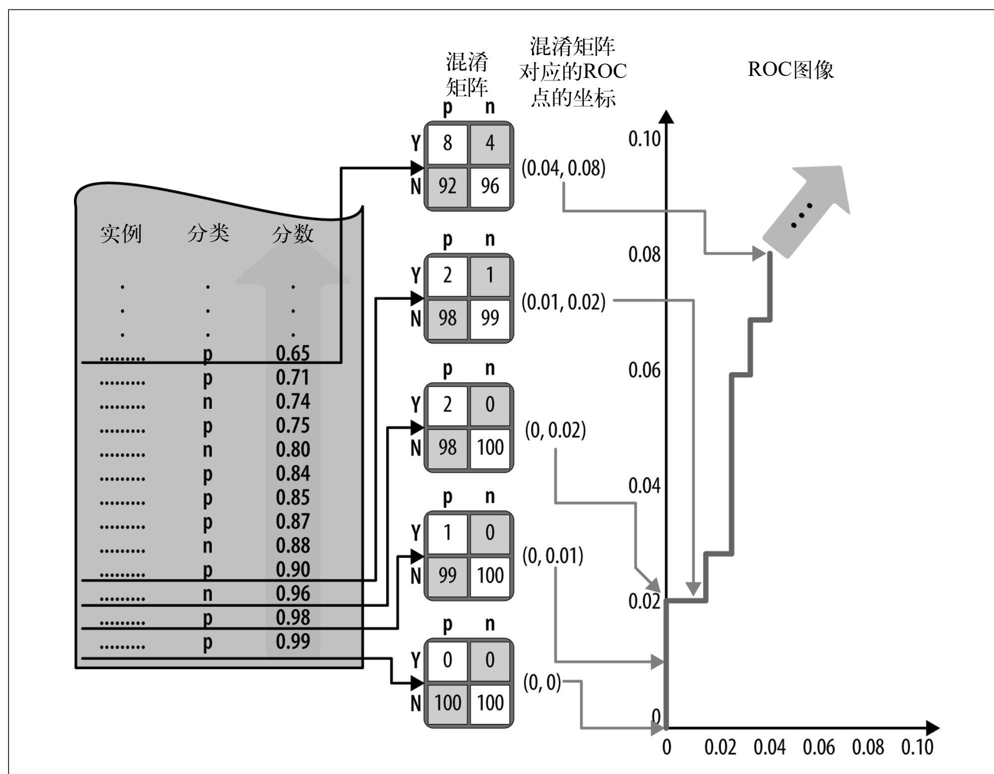


图 8-5：根据测试集数据构建 ROC“曲线”（其实是一个阶梯图）的图解。左侧的实例集包含了 100 个正实例和 100 个负实例，模型给每个实例分配分数，并将它们按分数从低到高排序。构建曲线时，先针对底部（所有实例都被判断为 N）的情况构建混淆矩阵。随着点的逐步上移，每次都有一个实例从 N 行移到 Y 行，从而得到新的混淆矩阵，而每个混淆矩阵都对应 ROC 空间中的一对（假阳性比率，真阳性比率）组合

ROC 图像的一个优点是可以把分类器的性能与分类器的使用场景区分开。也就是说，分类器的性能是与类别比例以及成本收益互不影响的。数据科学家在生成分类器的时候，可以在 ROC 图像中绘制其性能的对应点，因为该点的位置和模型的相对性能不会发生改变。在 ROC 图像中，虽然我们感兴趣的区域可能会随着成本、收益以及数据中不同类别比例的改变而发生变化，但是 ROC 曲线本身会保持不变。

Stein（2005）和 Provost & Fawcett（1997, 2001）展示了如何组合分类器运行条件（类的先验概率和惩罚系数），从而找到 ROC 曲线中我们感兴趣的区域。简单地说，我们可以把关于可能类别的先验概率的阈值的知识与关于决策的成本收益的知识结合起来，从而描绘一组能辨别该条件下应该选择哪个（或哪几个）分类器的切线。Stein（2005）在一个金融案例（贷款违约）中展示了如何通过这种方法来选择模型。

8.4 ROC曲线下面积

ROC 曲线下面积（AUC）是一个重要的统计量。顾名思义，这个统计量指的是分类器曲线下，以单位正方形的形式表示的面积，值域为 0 到 1。虽然 ROC 曲线比这个面积信息量更大，但是当我们需要一个数字来概括模型性能，或者对运行条件一无所知时，AUC 这个统计量更加有用。8.6 节将展示 AUC 统计量的使用方法，目前读者只需要知道，AUC 是一个能够很好地反映分类器预测效果的统计量。



技术说明：AUC 与秩和检验等价。后者是统计学中一种知名的排序方法（Wilcoxon, 1945）。在进行极小的代数转换后，AUC 还与基尼系数等价（Adam & Hand, 1999; Stein, 2005）。同时，这两个指标也都等价于随机选择的正实例比随机选择的负实例排序靠前的概率。

8.5 累积响应曲线和提升曲线

ROC 曲线是对模型的分类性能、类概率估计性能和评分性能进行可视化的常用工具。但是，如果你刚刚接触这些概念，对这一切都不熟悉，那么 ROC 曲线其实并不是最直观的可视化工具，对于最应该理解这个结果的企业利益相关者来说，尤其如此。数据科学家应该明白，与利益相关者进行清晰的交流，不仅是工作中的一项基本目标，还是构建正确模型（以及正确地建模）的基础。因此，我们或许还要考虑其他可视化评估框架，虽然它们可能没有 ROC 曲线那样多的优点，但是更直观。（对企业利益相关者而言，最重要的是要明白，那些为了交流而牺牲的理论细节有时也很重要，因此在特定环境下，我们也有必要展示一下比较复杂的可视化。）

可以替代 ROC 曲线的一个常用工具是“累积响应曲线”，虽然这两者联系密切，但是后者更为直观。累积响应曲线将命中率（或称**真阳性比率**；y 轴），即被**正确分类的正实例的比例**，作为目标群体占总体比例（x 轴）的函数，因此，从概念上来讲，当沿着被模型降序排列的实例列表下移时，被覆盖的目标群体的比例也在逐渐增大。如果过程顺利且模型性能良好的话，那么在列表顶端的目标群体中，实际为正的实例的比例将高于实际为负的实例比例。与 ROC 曲线相同，累积响应曲线图像中的对角线 $x = y$ 也代表随机性能。在这个例子中，我们可以清楚地感觉到：如果完全随机地选定 20% 的目标实例，那么这其中一定也包含了 20% 的正实例。任何位于对角线上方的分类器都比随机分类更有优势。



累积响应曲线有时也被称作**提升曲线**，因为它能用模型曲线（表示模型性能）向上远离对角线（表示随机分类器性能）的程度来展示模型的效果相对于随机选择的提升程度。但下文将继续称这些曲线为累积响应曲线，因为“提升曲线”也可以指提升度数值的曲线。

直观上，分类器的提升表示的是它相对随机预测结果的优势。提升度指分类器在列表中将正实例“提升”至负实例之上的程度，例如，假设一个列表中有 100 个用户，其中有一半已经离开公司（正实例），另一半仍留在公司（负实例），如果你从上向下浏览排序列表并停在中间位置上（目标群体占比为 50%），那么在浏览过的上半部分数据中，你认为会有多少正实例呢？如果列表数据的顺序是随机的，那么上半部分应该会含有一半正实例（0.5），因而提升值是 0.5/0.5 = 1；如果数据由有效的排序分类器进行了排序，那么列表的上半部分就会包含超过一半的正实例，从而使提升度大于 1；如果分类器是**完美**的，那么上半部分就会包含所有正实例（1.0），从而使提升度为 1.0/0.5 = 2。

图 8-6 描绘了 4 个样本分类器的累积响应曲线，图 8-7 则展示了这四个示例的提升曲线。

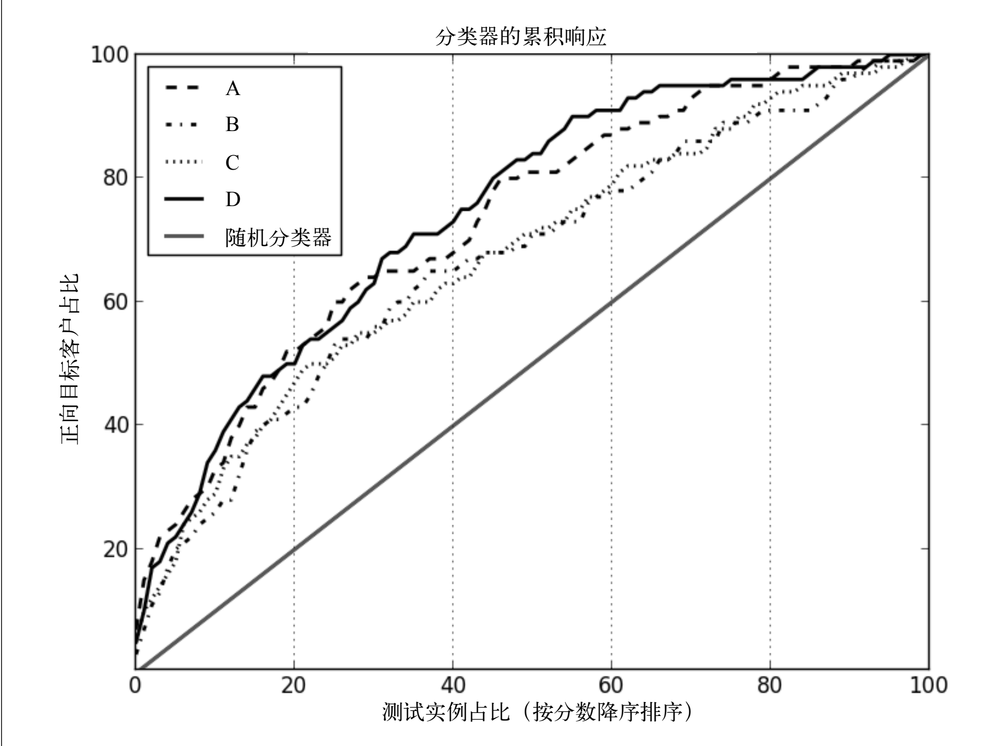


图 8-6：四个分类器示例（A~D）及其累积响应曲线

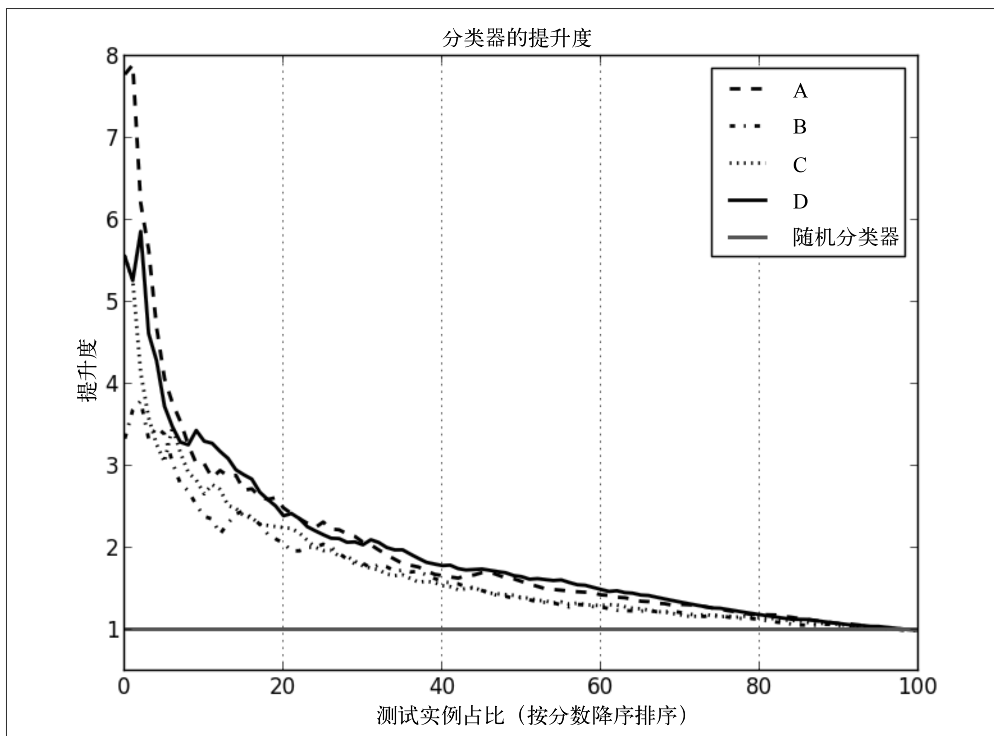


图 8-7：图 8-6 中的四个分类器（A~D）及其提升曲线

提升曲线的值其实是累积响应曲线在给定 x 点处的值除以对角线（$y = x$）在该点的值。提升曲线上 $y = 1$ 时，累积响应曲线的对角线将变为水平线。

有时你会听到类似“我们的模型有 2 倍（或 2X）提升”的说法，这表示在给定阈值的情况下（通常隐去不提），提升曲线表明模型选定的目标群体比随机选定的目标群体要好一倍。在累积响应曲线上，对应模型的**真阳性比率**是随机分类的性能曲线（对角线）的两倍（也可以按照其他基线计算提升）。提升曲线的 y 轴代表提升数值，x 轴代表目标群体所占比例（与累积响应曲线的 x 轴含义相同）。

在使用提升曲线和累积响应曲线时必须非常谨慎，因为目标群体中正实例的比例有时是未知的，或在测试数据中没有被准确代表。与 ROC 曲线不同的是，这两个曲线需要假设测试集中的目标类先验概率与要应用模型的目标群体中类的先验概率相同。这也是我们最初提到的简化假设之一，它可以让我们使用更加直观的可视化工具。

举个例子。在线上广告中，消费者响应某一条广告的基础比率可能非常小，一千万分之一（$1:10^7$）的比率也很常见。未响应者与响应者的比例为 1000 万比 1 对建模者来说是件很麻烦的事情，于是他们会对未响应者降低采样，从而创建更为平衡的数据集，以便建模和评估。这种做法在用 ROC 曲线来对分类器进行可视化时不会影响结果（因为上文提到，图中坐标轴仅对应一个类的比例），但在提升曲线和累积响应曲线中却不然——虽然曲线的基本形状仍然包含着大量的信息，但是坐标轴上的值之间的关系会失效。

8.6 示例：用户流失模型的性能分析

前几章已经介绍了不少评估方面的知识，包括很多重要的评估方法和评估模型中的各种问题。本节用一个应用示例研究将它们联系在一起，来展示不同评估方法的结果。这个示例依然来自被反复提到的手机用户流失问题，但本节将使用另一个（比前面章节中更为复杂的）流失数据集，它来自 2009 KDD CUP 数据挖掘大赛（http://www.kddcup-orange.com/）。在前面的示例中（见表 3-2 和图 3-18），我们之所以没有用到这个数据集，是因为为了避免泄露用户隐私，一些属性的名称和值已经被隐去。因此，利用它不仅意义不大，还可能会对讨论造成干扰。然而，我们可以用清洗过的数据来进行模型性能分析。以下内容来自该网站：

> KDD Cup 2009 提供了一个用法国手机公司 Orange 的大型营销数据库来预测用户行为的示例，其中包括预测用户变更供应商（用户流失）、购买新产品或服务（偏好）或购买推荐给他们的升级版本或附件（追加销售）的倾向。实现这些最实用的方法是，在客户关系管理系统（CRM 系统）中利用客户信息生成每个客户的分数
>
> 分数（模型的输出项）是对全部实例的待解释的目标变量（即流失、偏好或追加销售）的一种评估。而生成分数的工具能帮助算出给定总体的量化信息。我们通过输入描述实例的变量计算分数，然后在信息系统（IS）中基于不同场景使用这些分数，比如将客户关系个性化。

由于数据集被清洗得比较彻底，故而几乎没有值得探讨的内容了，但在这里还是要提一下实例偏斜的问题。数据集共包含 47 000 个实例，其中 7% 的实例为流失用户（正实例），剩下的 93% 则没有流失（负实例）。这样看来，其实数据集的偏度并不算大，但是出于后面将提到的某些原因，我们还是有必要提一下。

需要强调的是，这样做既不是为了提出解决问题的好方法，也不是为了表明哪个模型效果更好，而只是想把这个情景当作检验模型评估思想的平台，而且我们并没有花费什么功夫来调整模型性能。我们将训练和测试下面几个模型：分类树、逻辑回归和最近邻模型，还会用到一个被称作朴素贝叶斯的简单贝叶斯分类器（第 9 章会介绍它）。本节不会介绍模型的细节，所有的模型都是性能特征不同的“黑箱”。我们将用前几章介绍的评估技术和可视化技术来理解它们的特征。

先从一种非常简单的评估讲起：先用整个数据集训练模型，再用**同一个**数据集进行测试。我们也会测量模型的简单分类准确率。结果如表 8-1 所示。

表8-1：用完整的KDD Cup 2009用户流失问题训练和测试的四个分类器的准确率

模　型	准确率
分类树	95%
逻辑回归	93%
k- 最近邻	100%
朴素贝叶斯	76%

这里有几个显著的特点。首先，模型性能分布很广，从 76% 到 100%。另外，由于数据集的基础比率是 93%，因而任何分类器的最小准确率至少要大于该数值。但是奇怪的是，朴素贝叶斯的性能结果远小于该数值。而 *k*- 最近邻分类器的准确率达到了 100%，性能好得让人生疑。[5]

不过，模型的测试是在训练集上进行的，现在（学习过第 5 章之后）你已经知道这样的数字并不可靠，甚至完全无意义，它更像是反映分类器能在多大程度上记忆（过拟合）训练数据的指标。因此我们不必深入研究这些数字，而应该用相互独立的训练集和测试集重新评估模型。尽管可以简单地把数据集分成两半，但在这里我们选用 5.6 节中的交叉验证方法，因为采用这个方法不仅能对数据集进行适当划分，还能对模型结果的变化进行度量。结果如表 8-2 所示。

表8-2：10重交叉验证后，四个分类器应用于KDD Cup 2009用户流失问题时的准确率和AUC值

模　　型	准确率	AUC
分类树	91.8 ± 0.0	0.614 ± 0.014
逻辑回归	93.0 ± 0.1	0.574 ± 0.023
k- 最近邻	93.0 ± 0.0	0.537 ± 0.015
朴素贝叶斯	76.5 ± 0.6	0.632 ± 0.019

表格中的每个数值都是 10 重交叉验证的均值加减（“±”）标准差的形式。其中标准差可以认为是一种“合理性检查”：过大的标准差意味着测试结果很不稳定，这可能是由各种问题导致的，比如数据集过小，或者模型与问题的一部分不匹配。

所有准确率都有明显下降，朴素贝叶斯除外（它依旧低得古怪）。因为模型的标准差皆比均值小得多，所以模型性能的标准差不大，而这是我们希望看到的情况。

最右边一栏中是 ROC 曲线下面积值（通常简称为 AUC）。8.4 节简要讨论了 AUC 测度，我们知道这是一种很好的关于评估分类器预测效果的统计量，值域为 0 到 1。AUC 为 0.5 意味着模型是随机预测（分类器完全无法区分正实例和负实例），AUC 为 1 则意味着分类器能够完美区分二者。而准确率就不是一个非常恰当的度量方法，其原因之一是当数据集偏斜时，这个指标会使人误解，正如这一节讨论的示例所示（负实例占 93% 而正实例只占 7% 的情况）。

我们曾在 5.3 节中介绍了拟合曲线，并将其作为检验模型是否存在过拟合的方法。图 8-8 展示了根据用户流失问题构建的分类树模型对应的拟合曲线。拟合曲线的基本思想是，模型越复杂，它对数据的拟合就越接近，但到了某个点，模型会开始只单纯记忆特定训练集的特征，而不是学习总体的普遍特征。拟合曲线描绘的是模型复杂度（本例中指树的节点数）与模型的性能测度（本例中指 AUC），后者由两个数据集（训练数据集和单独的保留数据集）计算得出，当保留数据集上模型的性能开始下降时，过拟合就产生了。图 8-8 的确符合这种一般规律。[6] 分类树的确**存在**过拟合问题，其他模型可能也一样。图中“甜蜜点”出现在节点数为 100 处，超过该数值时，模型在保留数据集上的性能就会下降。

注 5：虽然乐观是好事，但根据数据挖掘的经验法则，在现实问题中任何完美的结果都不可信。

注 6：注意，*x* 轴做了对数化处理，因此图右侧看起来比较拥挤。

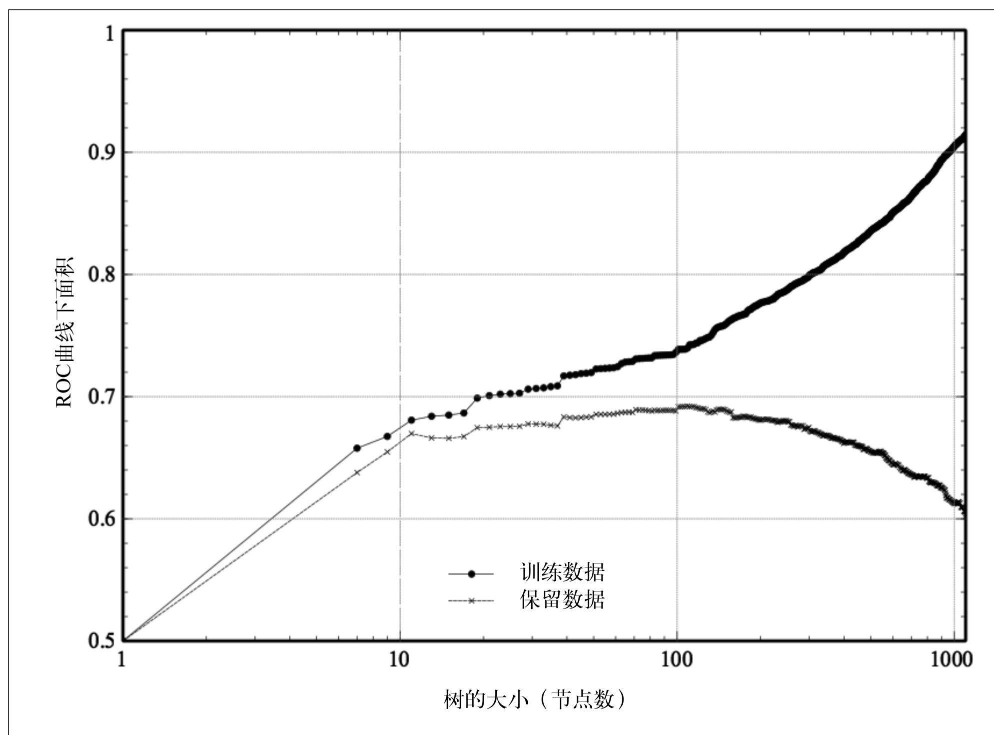


图 8-8：根据用户流失数据构建的分类树的拟合曲线，其中 ROC 曲线下面积（AUC）随模型复杂度的改变而改变。训练数据集上模型的性能（上方的曲线）持续提升，而保留数据集上模型的性能先达到顶峰，随即下降

请回顾一下表 8-2 中模型的比较指标，因为这些值在保留数据集上做了合理谨慎的评估，所以比较可靠。然而，这里面也确实存在一些问题。关于 AUC 的值有两点很值得讨论。一点是，这些模型的 AUC 值都一般。其实这在实际应用场景中并不少见，这或是因为数据集中可挖掘的信息很少，或是因为数据科学问题在较为简单的问题解决之后才构建起来。由于用户流失问题比较复杂，因而模型的 AUC 值较低并不奇怪。即使 AUC 的评分一般，模型解决商业问题的结果也可能会很好。

第二个值得关注的点是朴素贝叶斯模型。如表 8-2 所示，在这组模型中，它的准确率**最低**，AUC 值却**最高**，这是为什么呢？请比较一下朴素贝叶斯模型（AUC 值最高而准确率最低的）的混淆矩阵和应用于同一数据集的 *k*- 最近邻（AUC 值最低而准确率最高）模型的混淆矩阵。下面是朴素贝叶斯模型的混淆矩阵。

	p	n
Y	127 (3%)	848 (18%)
N	200 (4%)	3518 (75%)

下面是应用于同一数据集的 *k*- 最近邻模型的混淆矩阵。

	p	n
Y	3 (0%)	15 (0%)
N	324 (7%)	4351 (93%)

可以看出，*k*- 最近邻模型几乎没有预测到流失用户，因为 Y 列几乎为空。换句话说，这个模型更像一个基础比率分类器，总准确率恰好约为 93%（基础比率）。然而朴素贝叶斯分类器犯的错误更多（因此准确率更低），但能够辨别出更多的流失用户。图 8-9 展示了交叉验证过程中一个典型折叠的 ROC 曲线，注意，这里对应朴素贝叶斯模型和分类树模型的曲线比其他曲线更为“弯曲”，这表示两者有着预测优势。

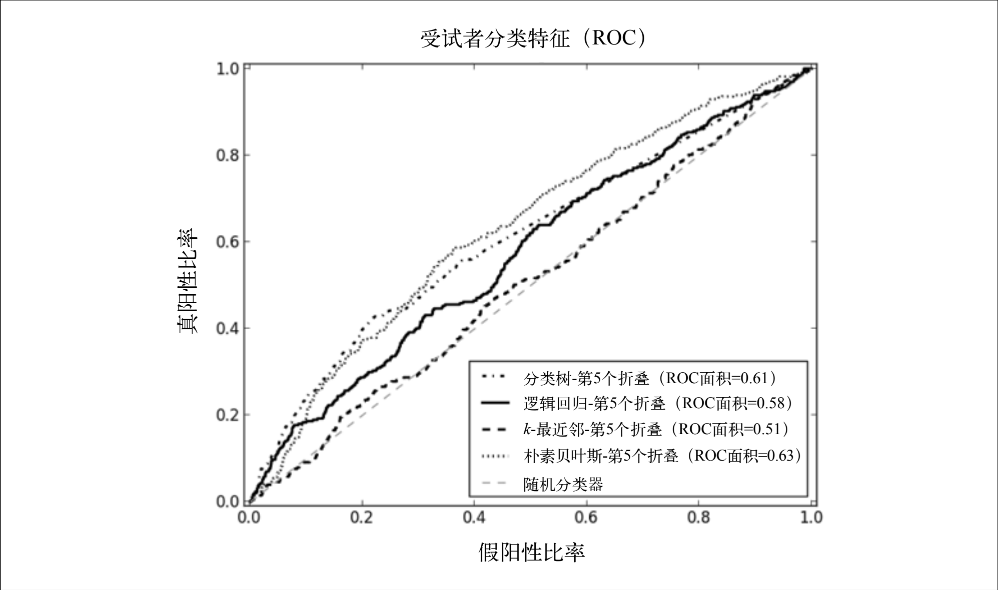


图 8-9：预测用户流失问题的分类器在交叉验证的一个折叠中的 ROC 曲线

正如前面所提到的，虽然 ROC 曲线有许多非常好的技术性质，但是不容易理解，像“弯曲”程度和相对预测优势就很难用肉眼识别。因此，提升曲线和利润曲线有时更加实用，下面我们将对这两个指标进行分别介绍。

因为提升曲线有着不需投入成本的优势，所以我们先从它讲起，请看图 8-10。

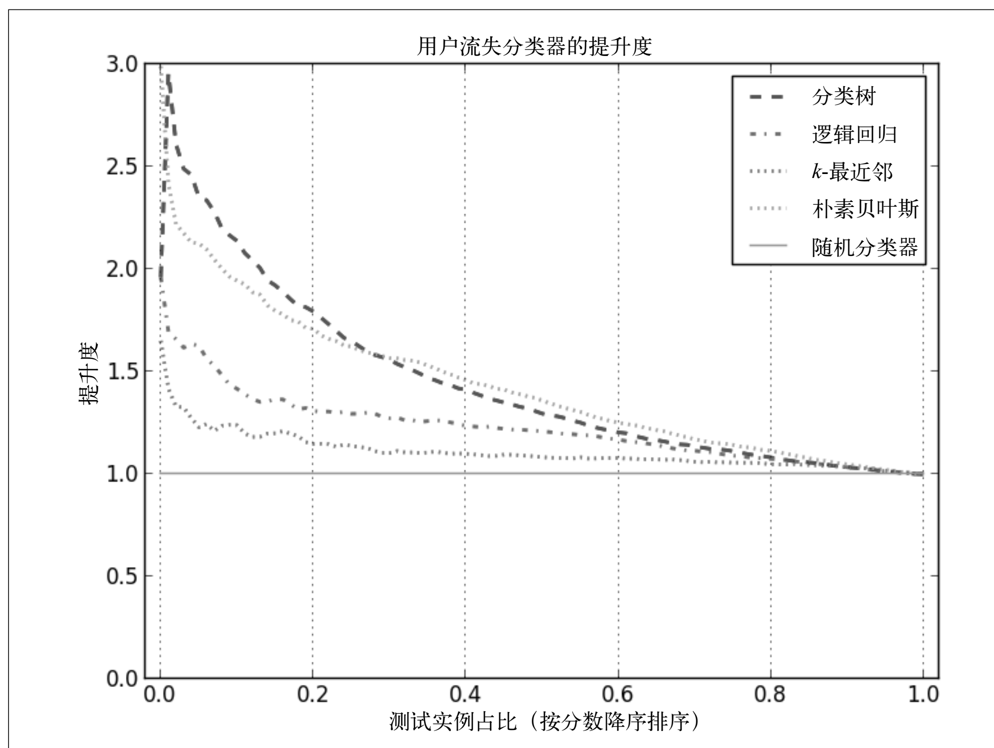


图 8-10：用户流失问题的提升曲线

这些曲线均根据 10 重交叉验证的结果取了均值，它们通常很早达到最高值，然后逐渐减小到随机性能处（提升度 = 1）。分类树模型和朴素贝叶斯的性能都非常好，分类树模型在目标群体比例小于等于 25% 时占优势，之后朴素贝叶斯模型更占优势。*k*- 最近邻和逻辑回归模型则性能相对较差，且没有占优势的区域。通过这张图像可以知道，如果目标群体的比例小于等于 25%，那么我们应选择分类树模型，否则应选择朴素贝叶斯模型。由于提升曲线对类比例比较敏感，因此，如果流失用户和未流失用户的比例改变，那么这些曲线也会随之改变。



有关组合分类器的一则说明

在观察这些曲线时，你可能会问："如果分类树模型在目标群体比例小于等于 25% 时最好，之后朴素贝叶斯模型最好，那么为什么不在前 25% 时选用前者，然后换成后者呢？"这是个好想法，但这样你可能无法充分利用这两种分类器。简单地说，是因为两者的顺序不同，所以简单地各选择两者中的一部分并不能达到最优结果。评估曲线仅对单个模型有效，而如果将模型组合混用，它就不再起作用了。

但我们可以有原则地将分类器进行组合，从而使组合后的分类器的性能超过任何单个分类器。我们把这样的组合叫作"集成"，12.5 节将对其进行讨论。

提升曲线虽然能够展示每个模型的相对优势，但**并不能**展示每个模型带来的收益，甚至不能展示是否会取得收益。想要实现后两个需求，就要使用利润曲线，因为利润曲线的假设包含成本收益，并且能展示其期望值。

请暂时忽略手机用户流失问题中的细节（将在第 11 章继续讨论）。为了让数据集更加有趣，我们将制作两套有关成本收益的假设。第一种情况下，假设每份优惠的成本是 3 美元、毛收益为 30 美元，因此真阳性实例的净利润是 27 美元，假阳性实例的净损失是 3 美元。这种情况下利润率为 9 : 1，其利润曲线如图 8-11 所示。分类树模型对最高的阈值而言是最合适的，而朴素贝叶斯模型则在剩下的阈值上占优势。这种情况下，最大收益在目标群体约占总体的 20% 时实现。

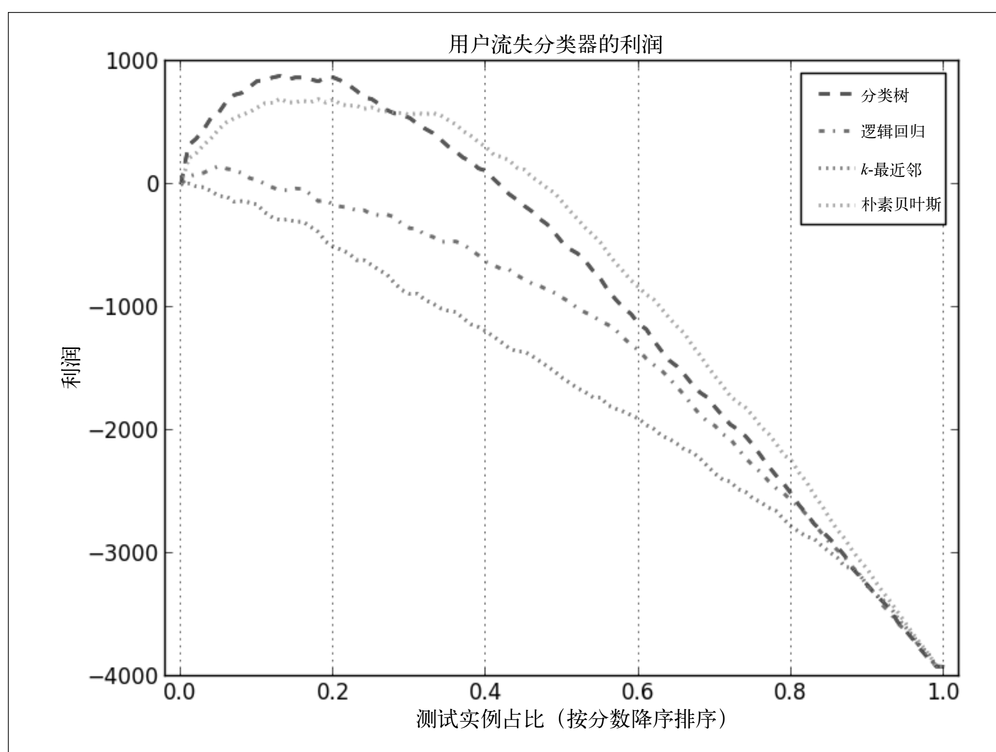


图 8-11：根据用户流失问题构建的 4 个分类器的利润曲线，假设收益与成本之比为 9 : 1

在第二种情况下，假设每份优惠的成本仍为 3 美元（因此假阳性成本未改变），但是毛收益提升至 39 美元，因此真阳性的净利润提升到了 36 美元，利润率为 12 : 1，则利润曲线如图 8-12 所示。你可能已经预料到了，这种情况下的最大收益比前一种的更高。更重要的是，它能够展示不同的利润**最大值**，其一是分类树模型在目标群体占 20% 时达到的，其二是朴素贝叶斯模型在目标群体占 35% 时达到的，后者比前者略高。然而，在两幅图中，分类树模型和逻辑回归模型的交叉点出现在同一个位置（约在目标群体占 25% 时）。这说明了利润曲线对成本收益的特定假设的敏感性。

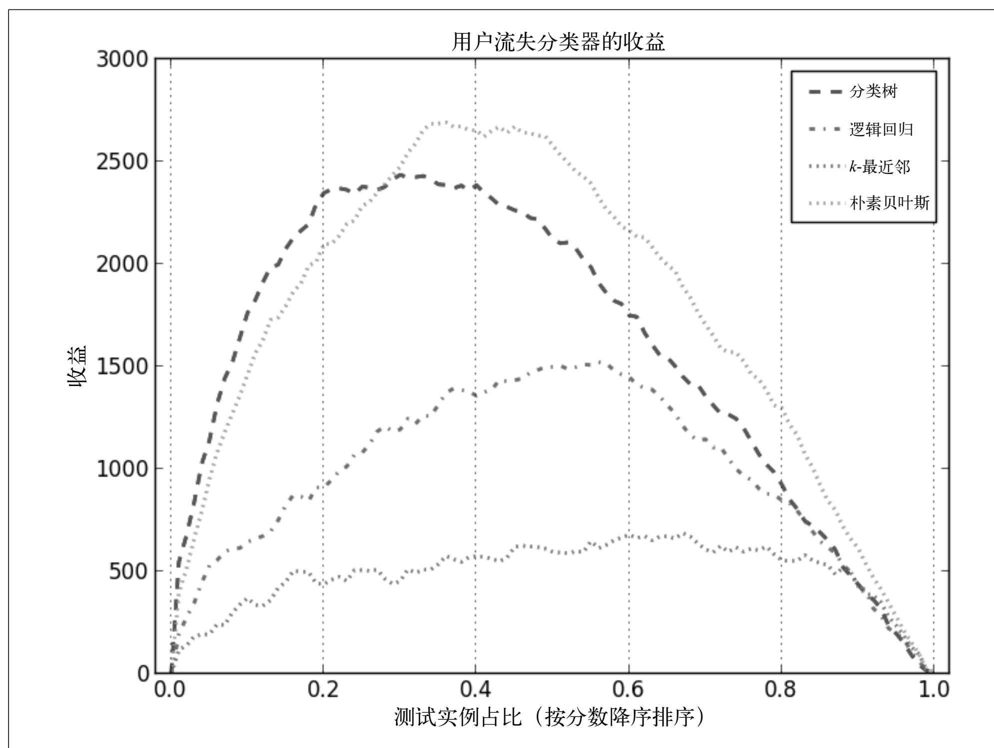


图 8-12：根据用户流失问题构建的 4 个分类器的利润曲线（与图 8-11 相比）假设利润率更高，为 12：1

本节结束之前，需要再强调一下，展示这些图像仅仅是为了阐述模型评估的不同技术。我们既没有花时间去调整模型的归纳方法，也没有对这些模型的优缺点及其是否适用于用户流失预测问题下结论，而只是慎重地计算了一系列分类器的性能，以阐述这些图像是怎样比较这些分类器的。

8.7 小结

数据科学家工作中一个非常重要的部分就是对模型进行恰当评估，并且把评估结果传达给利益相关者。虽然做好这项工作需要大量经验，但是为了减少意外情况并且满足所有相关人士的期望，这项工作又十分关键。而模型结果可视化是评估任务中重要的一部分。

在建模的时候，我们可能很有必要，甚至是必须，通过多种方式来调整训练样本，但在评估模型的时候，却必须选择能够反映原始总体分布的数据集，只有这样，才能保证模型结果反映出真实的结果。

如果决策的成本和收益能被明确规定，那么数据科学家就可以针对每个模型计算每个实例的期望成本，然后选择拥有最优值的模型。一些情况下，基本的**利润**图像足以比较各模型在一系列条件下的优劣。这些图像对欠缺数据科学背景的利益相关者来说很好理解，因为它们把模型性能用基本“底线”，即成本或收益的形式表示了出来。

收益图像的一个缺点是，它要求模型的运行条件已知且明确，然而，在许多现实问题中，运行条件并不准确或会随时改变，于是数据科学家面临着大量的不确定性问题。这种情况下，其他图像会更占优势。如果成本和收益无法明确，但类别比例不大会发生改变，那么我们通常会选用**累积响应曲线**或**提升曲线**，两者都能展示分类器的相对优势，而且不受优势的值（货币价值等）的影响。

最后，ROC 曲线也是一种重要的可视化工具。虽然使用者需要有一定经验才能较好地解读它，但它能将模型性能与其运行条件分离，从而表现出每个模型所做的权衡。

机器学习领域和数据挖掘领域中的大量工作都涉及通过比较分类器来证明学习算法的优越性。因此，介绍分类器比较方法的文章有很多。如果读者感兴趣，那么不妨从 Thomas Dietterich（1998）的文章"Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms"和图书 *Evaluating Learning Algorithms: A Classification Perspective*（Japkowicz & Shah, 2011）读起。

第 9 章

证据和概率

基本概念：根据贝叶斯法则对证据进行显式组合；基于条件独立假设进行概率推理

示例方法：朴素贝叶斯分类；证据提升度

到目前为止，本书已经探讨了好几种方法，它们都可以用数据得出实例的某些未知量，比如对实例的分类。现在本书要探讨另一种这样的方法。你可以把你对实例的了解视作支持或反对不同目标变量值的**证据**，而对实例的了解则可以表示为实例的特征。如果你知道每个特征所提供的证据的强度，那么就能应用原则性方法，从概率上合并证据，从而得到有关目标变量值的结论。证据的强度将根据训练数据确定。

9.1 示例：向线上目标用户投放广告

为了便于说明，请考虑分类的另一种商业应用——根据用户浏览过的页面，对目标用户投放线上展示广告。作为消费者的我们，已经对网页上貌似免费的大量信息和服务习以为常。当然，所谓的"免费"往往建立在线上广告存在（或很有希望带来）收益的基础上，这与广播电视的"免费"大同小异。此处考虑的是**展示广告**，即出现在我们正在阅读或消费的网页的顶部、两侧或底部的广告。

展示广告与搜索广告（比如，展示在谷歌搜索结果中的广告）不同。两者的一个重要区别在于：在大多数页面中，用户通常不会输入任何与他真正想找的内容有关的文字。因此，我们需要基于其他类型的推断来判断一支广告的目标用户。过去的几章已经探讨过一种推断方法：通过实例的特征值来推断其目标变量值。因此，我们可以将该技术用于推断某个用户是否对某个广告感兴趣。本章将介绍看待此问题的另一种方法，该方法不但广泛适用，而且非常易于应用。

先让我们更精确地定义一下这个精准广告的问题。实例是什么？目标变量是什么？特征又

是什么？如何获取训练数据？

假设我们在为一家大型内容供应商（“出版商”）工作，该供应商包含的内容种类繁多，拥有许多线上用户，且有许多机会可将广告展示给这些用户。比如，雅虎（Yahoo!）就拥有大量由广告支持的网页“资产”，或者说不同的“内容块”。另外，目前（本书完成前），雅虎已同意收购 Tumblr，一个包含超过 1 亿个博客和 500 亿余篇博文的博客平台。其中每一个博客都可以视作为感兴趣的用户提供信息的“内容块”。与之相似，Facebook 用户留下的每一个“赞”也可以视为体现用户喜好的证据，这同样有助于精准投放广告。

为简化问题，假设有一个广告活动，需要针对一些访问我们网站的线上用户进行投放。该广告的主题是高档连锁酒店 Luxhote，目的是让人们来订房间。以前我们开展过一次活动，当时随机选择了线上用户。而现在我们想有针对性地投放广告，以期花在广告曝光量上的单位资金能带来更多的订单。[1]

因此，我们把每个用户视作一个实例，而目标变量则为“该用户是否在观看 Luxhote 广告后一周内预订了 Luxhote 的房间”。借助神奇的浏览器 cookie[2]，通过与 Luxhote 合作，可以观察哪些用户订了 Luxhote 的房间。为便于训练数据，我们把每个用户的该目标变量设为二值型。在实际应用中，我们将会估计用户在观看广告后订房的概率，然后根据预算情况，选择概率最高的那些用户作为目标。

还有一个关键问题有待解决：用于描述用户的特征是什么？只有有了这些特征，才能辨别出那些更优质的 Luxhote 潜在客户。对本例而言，我们要通过浏览器 cookie 或其他途径，用某用户浏览过（或点“赞”）的一系列内容块来对其进行描述。内容的类型有很多，包括金融、体育、娱乐和美食博客等。我们可以选择几千个热度很高的内容块，也可以选择上亿个，我们相信这些内容里的一部分（比如金融类博客）更容易被 Luxhote 的优质潜在用户访问，另一部分则可能性较低（比如，拖拉机拉力赛粉丝网页）。

然而，在本例中，我们并不想依赖这种对内容的假设，也没有手动估计每个内容块证据可能性的资源。况且，人类虽然能很好地用知识和常识辨别证据支持结论还是反对结论，但是在准确估计证据**强度**方面奇差无比。而我们希望历史数据不仅能用于判断证据的方向（支持或反对），还能用于估计证据的强度。接下来，本章将介绍一个适用范围极广的框架，它不只适用于证据评估，也适用于结合证据以估计类成员可能性（此处指用户在观看广告后订房的可能性）。

事实证明，许多问题都与该示例的模式相符合：在分类或类概率估计问题中，每个实例都由一组证据描述，而这些证据可能提取自一个很大的、包含所有可能证据的集合。举个例子，文本文档分类就完全符合该模式（第 10 章将探讨）。每篇文档都是一组单词的集合，而这些单词来自一个庞大的词汇表。每个单词可能都会提供一些支持或反对分类的证据，而我们需要将这些证据结合起来。接下来要介绍的技术正是许多垃圾邮件检测系统所使用的：一封电子邮件即一个实例，其目标类别分为**是垃圾邮件**和**不是垃圾邮件**，而其特征则是邮件中的单词和符号。

注 1：广告**曝光量**指的是广告展示在页面上的数量，不考虑用户是否点击它。

注 2：浏览器会与所访问的站点交换少量信息（即“cookie”）并存储站点特有的信息，以便以后访问同一站点时直接提取。

9.2 根据概率合并证据



下文含有较多数学内容

为探讨根据概率合并证据的思想，本书需要引入一些概率记号。读者无须掌握（或记住）概率论的知识，因为这些记号非常直观，而本书的讨论也不会超出基本概念的范围。这些记号能使本书的讲述更加精确。下文的数学知识看起来可能较多，但你会发现它们其实都很简单。

我们关注的是用户在观看广告后订房的概率之类的量。而实际上，我们需要更明确一些。用户是特定的吗？还是任意用户都可以？首先考虑后者。如果将广告展示给任意一个用户，那么此人订房的概率是多少？由于这是我们关心的分类问题（classification），因而将其记作量 C，并将事件 C 发生的概率记为 $p(C)$。$p(C) = 0.0001$ 的含义是：如果随机地将广告展示给用户，那么我们认为 10 000 个用户中会有一个订房。[3]

然后我们要计算，给定某个证据 E 后事件 C 发生的概率，其中证据 E 可以表示某个**特定**用户访问了一组网站。我们将这个概率记作 $p(C|E)$，读作“给定 E 后 C 的概率”或“在 E 的条件下 C 的概率”。这是一种条件概率，其中“|”有时也叫作“条件条”。$p(C|E)$ 应随证据 E 的改变（本例中指的是访问过的网站组）而改变。

如上文所述，我们会用一些标注数据（如在随机投放广告活动中收集的数据）把证据 E 的不同集合与不同概率关联起来。但这会导致一个关键问题：对证据 E 的任何特定集合而言，我们可能无法找到足够多的证据集合与之完全相同的案例，以致无法确定地推出类成员概率。事实上，我们甚至可能根本找不到这样的证据集合！试想在本例中，如果要考虑上千个网站，那么，训练数据中某用户的访问模式，与将来的某用户**完全**相同的概率是多少？恐怕是无穷小。因此，我们应该分别考虑不同的证据，然后将它们合并起来。为了更深入地探讨这一点，需要介绍一些关于合并概率的概念。

9.2.1 联合概率与独立性

假设有两个事件 A 和 B，如果 $p(A)$ 和 $p(B)$ 已知，那么是否可以计算两个事件同时发生的概率？该概率被称为**联合概率**，记作 $p(AB)$。

在一种特定情况下，我们能计算该联合概率：当事件 A 和事件 B 相互独立时。两者相互独立指的是，其中任意一个事件都不包含另一事件的任何概率信息。一个解释独立性的典型例子是掷均匀骰子：即使已知第 1 次掷骰子的点数，也无法确定第 2 次掷骰子的点数。如果事件 A 是“第 1 次掷骰子的点数为 6”，事件 B 是“第 2 次掷骰子的点数为 6”，那么 $p(A) = 1/6$、$p(B) = 1/6$，而且，重要的是，虽然已知第 1 次掷骰子的点数为 6，但是 $p(B)$ 依旧是 1/6。该例中的两个事件就相互独立，而这种情况下，$p(AB) = p(A) \cdot p(B)$——我们可以通过将独立事件的概率相乘来计算“联合”事件 AB 的概率。此处的 $p(AB) = 1/36$。

注 3：这只是一个示例，不是对任何广告都适用的合理响应率。比如对行业外的人士来说，线上广告所贡献的购买率看起来通常非常低。不过，投放线上广告的成本通常也非常低。

但这不是计算联合概率的一般方法。如果难以理解，那么不妨考虑掷魔术骰子的例子。我的口袋里有 6 个魔术骰子，每个魔术骰子上有 1~6 中的一个数字，每一面的数字都相同。我随机从口袋里取出一个骰子，然后掷两次。本例中，$p(A) = p(B) = 1/6$（因为我从口袋中取出每一个骰子的概率相等），但 $p(AB)$ 也为 1/6，因为这两个事件完全不独立！如果第 1 次掷骰子的点数为 6，那么第二次也应为 6（反之亦然）。

考虑到事件间的依赖性，合并概率的一般公式如下。

公式 9-1：用条件概率表示的联合概率

$$p(AB) = p(A) \cdot p(B|A)$$

该公式读作："A 和 B 的概率等于 A 的概率乘以 B 在 A 的条件下的概率"。换句话说就是：如果已知 A，那么 B 的概率是多大？不要着急，请确保完全理解这些概念。

可以用以上两个骰子示例来解释该公式。在两者独立的示例中，因为知道 A 的信息并不会让我们了解 B，所以 $p(B|A) = p(B)$，因此套入上述公式，我们简单地将单个概率相乘即可；而魔术骰子的示例中，$p(B|A) = 1.0$，因为如果第 1 次掷骰子的点数为 6，那么第 2 次的点数一定也为 6，所以 $p(AB) = p(A) \cdot 1.0 = p(A) = 1/6$，符合我们的预期。

通常，事件可以完全独立、完全不独立或介于二者之间。如果两事件并非完全独立，那么已知其一，另一事件的概率就会受到影响。但 $p(AB) = p(A) \cdot p(B|A)$ 在所有情况下都适用。

我们之所以探讨这些细节，是由于一个非常重要的原因——该公式是数据科学中（确切地说，是所有的科学中）最著名的公式之一的基础。

9.2.2　贝叶斯法则

你可能注意到，公式 $p(AB) = p(A)p(B|A)$ 中 A 和 B 的顺序看起来非常随意。的确如此。我们这样写也无妨：

$$p(AB) = p(B) \cdot p(A|B)$$

这意味着：

$$p(A) \cdot p(B|A) = p(AB) = p(B) \cdot p(A|B)$$

从而：

$$p(A) \cdot p(B|A) = p(B) \cdot p(A|B)$$

两边同时除以 $p(A)$，得到：

$$p(B|A) = \frac{p(A|B) \cdot P(B)}{p(A)}$$

现在假设 B 是我们感兴趣且想要评估概率的假设，A 是已观测到的证据，将假设重命名为 H、证据重命名为 E，得到：

$$p(H \mid E) = \frac{p(E \mid H) \cdot p(H)}{p(E)}$$

这便是著名的**贝叶斯法则**，以牧师 Thomas Bayes 的名字命名，他在 18 世纪推导出了该法则的一个特例。贝叶斯法则表明，我们可以利用假设 H 条件下证据 E 的概率，以及假设 H 与证据 E 的无条件概率，计算出在给定证据 E 的条件下假设 H 的概率。



贝叶斯方法

贝叶斯法则与仔细思考条件独立性这一基本概念相结合，构成了大量更为高级的数据科学技术（本书中并未提及）的基础。这些技术包括贝叶斯网络、概率主题模型、概率关系模型、隐马尔可夫模型、马尔可夫随机场等。

重要的是，后三个量会比最受关心的量（即 $p(H|E)$）更容易确定。为了便于理解，读者不妨考虑一个（简化后的）医疗诊断的例子：假如你是一名医生，接诊了一位身上长着红斑的患者，你推测（假设）他是长了麻疹。我们要计算在给定证据（E= 红斑）的情况下，假设诊断（H= 麻疹）正确的概率。为直接估计 p(麻疹 | 红斑)，我们需要考虑所有可能致使患者长红斑的原因，以及麻疹在其中所占的比例，而即使是学识最为广博的医师也不可能做到这一点。

然而，我们可以用贝叶斯法则公式右侧的式子来估计这个量。

- $p(E|H)$ 是得了麻疹的人长红斑的概率。传染病专家应该知道这一点，或者能够相对准确地进行估计。
- $p(H)$ 是患者得麻疹的概率，不将任何证据考虑在内，而仅是总体中麻疹的患病率。
- $p(E)$ 是证据的概率，即患者长红斑的概率。同样，这仅是总体中红斑的患病率，只需观察和计数，而无须对不同根本原因进行复杂推理。

贝叶斯法则使得对 $p(H|E)$ 的估计变得简单多了。我们需要三条信息，但这三条信息比初始值更容易估计。



$p(E)$ 可能依然不容易计算，但在许多情况下，我们并不需要计算该值，因为我们对比较相同证据条件下不同假设的概率更感兴趣。下文将继续对其进行探讨。

9.3 将贝叶斯法则应用到数据科学中

现在，贝叶斯法则在数据科学领域的关键性应该已经显而易见了。确实，数据科学有极大一部分内容基于“贝叶斯”方法，而贝叶斯方法的核心推理又基于贝叶斯法则。但是全面地描述贝叶斯方法远超出了本书的范围，因此本章将仅介绍最基本的概念，然后展示它们在最基本的贝叶斯技术中的应用（后者也得到了广泛应用）。请再次重写贝叶斯法则，但这次回到分类问题。我们暂时用“$C=c$”代表目标变量取值为 c，以强调该法则应用于分类。

公式 9-2：分类中的贝叶斯法则

$$p(C=c\,|\,\boldsymbol{E})=\frac{p(\boldsymbol{E}\,|\,C=c)\cdot p(C=c)}{p(\boldsymbol{E})}$$

公式 9-2 中共有 4 个量，左侧的量是我们需要估计的，在分类问题中，这就是在考虑证据 $\boldsymbol{E}$（即特征值**向量**）之后，目标变量 C 取值为 c 的概率，称作**后验概率**。

贝叶斯法则将后验概率分割成公式右侧的三个量。我们希望能通过数据计算出这些量。

(1) $p(C=c)$ 是类别的**先验**概率，即我们在看到证据前给类别分配的概率。在贝叶斯一般推理中，该概率可能来自多种途径，首先，“主观”先验，即某个决策者基于其所有的知识、经验和观点得出的信念；其次，基于先前在其他证据上应用的贝叶斯法则得出的“先验”信念；最后，根据数据推断出的无条件概率。下文介绍的具体方法采用了最后一种方法，将 c 的“基础比率”（整个总体中 c 的流行率）作为**类先验**。该指标即类 c 在所有实例中的百分比，可以根据数据很轻易地计算得出。

(2) $p(\boldsymbol{E}|C=c)$ 是在类 C=c 的条件下，证据 $\boldsymbol{E}$（被用于对实例分类的特征）的概率。你可以把这看作一个“衍生”问题：如果世界（数据生成过程）中创建出一个 c 类的实例，那么它与 $\boldsymbol{E}$ 相似的概率是多少？我们可以通过计算数据中 c 类实例含有特征向量 $\boldsymbol{E}$ 的比例得知答案。

(3) 最后，$p(\boldsymbol{E})$ 是证据的概率，即特征向量 $\boldsymbol{E}$ 在所有实例中的普遍程度。该指标可以通过计算特征向量 $\boldsymbol{E}$ 在所有实例中出现的百分比得知。

估计出训练数据中的这三个值后，我们可以计算后验概率 $p(C=c|\boldsymbol{E})$ 的估计值，并将其用于具体实例。该后验概率可以直接作为类概率的估计，还可能会（如第 7 章所述）与成本收益相结合，也可以作为对实例进行排序的评分（比如，判断最有可能对广告做出响应的用户）。我们也可以把不同 c 值下 $p(C=c|\boldsymbol{E})$ 所取的最大值作为分类结果。

然而，我们又回到了上文提到的主要问题，即让我们无法将公式 9-2 直接应用于数据挖掘的问题。假设 $\boldsymbol{E}$ 是一个属性值为 $<e_1, e_2, \cdots, e_k>$ 的普通向量，是一个可能很大的、具体的条件集合。若想对其直接应用公式 9-2，必须事先知道 $p(e_1 \wedge e_2 \wedge \cdots \wedge e_k|c)$[4] 形式的 $p(\boldsymbol{E}|c)$。这一点很特殊，而且极难度量，数据中可能并没有完全符合测试集中特定 $\boldsymbol{E}$ 的具体实例，即使有，我们也很可能无法从中确定地估计出概率。

数据科学中的贝叶斯方法通过假设概率独立性来解决这样的问题。解决这种复杂问题的最常用方法是对独立性做非常强的假设。

9.3.1　条件独立和朴素贝叶斯

回忆上文的独立概念：两事件相互独立，意味着已知其中之一，不会得知另一事件的概率信息。我们来稍微扩展一下这个概念。

条件独立的概念与之相同，但使用的是条件概率。根据目的，我们将把实例的类作为条件（因为在公式 9-2 中，我们需要寻找**给定**类中证据的概率）。条件独立与上文中讨论过的无条件独立直接相似。特别地，在不做独立性假设的情况下，为合并概率，我们需要用到用 $|C$ 条件增强后的公式 9-1：

$$p(AB|C) = p(A|C) \cdot p(B|AC)$$

注 4：“$\wedge$”表示“与”。

然而，如上文所述，如果我们假设 A 和 B 在给定 C 的情况下条件独立[5]，就能更轻松地合并概率：

$$p(AB \mid C) = p(A \mid C) \cdot p(B \mid C)$$

由此，我们根据数据计算概率的能力发生了巨大的改变，尤其是对公式 9-2 中的条件概率 $p(\boldsymbol{E}|C=c)$ 而言。假设对于给定的类而言，变量互相条件独立，也就是说，在特征向量 $p(e_1 \wedge e_2 \wedge \cdots \wedge e_k \mid c)$ 中，给定类 c，每个 e_i 都与其他 e_j 相互独立。为简化描述，只要不会招致误解，我们就用 c 替代 $C=c$。

$$\begin{aligned} p(\boldsymbol{E} \mid c) &= p(e_1 \wedge e_2 \wedge \cdots \wedge e_k \mid c) \\ &= p(e_1 \mid c) \cdot p(e_2 \mid c) \cdots p(e_k \mid c) \end{aligned}$$

每个 $p(e_i|c)$ 都能通过数据直接计算得出，因为我们只需计算 c 类中个体特征 e_i 出现的次数所占比例即可，而不需要寻求与之完全匹配的特征向量。这样的特征向量 e_i 可能会出现很多次[6]。将其与公式 9-2 相结合，就得到了**朴素贝叶斯方程**，如公式 9-3 所示。

公式 9-3：朴素贝叶斯方程

$$p(c \mid \boldsymbol{E}) = \frac{p(e_1 \mid c) \cdot p(e_2 \mid c) \cdots p(e_k \mid c) \cdot p(c)}{p(\boldsymbol{E})}$$

这是**朴素贝叶斯分类器**的基础。该分类器能通过估计新个体属于每个分类的概率，对其进行分类，并输出概率最高的类。

以下是两段技术细节。此刻你可能已经发现了公式 9-3 中的分母含有 $p(\boldsymbol{E})$，你可能会说："如果我的确看懂了的话，那么这个值的计算对我来说不是和 $p(\boldsymbol{E}|C)$ 一样难吗？"可实际上，$p(\boldsymbol{E})$ 一般不需要计算，原因如下：首先，我们如果对分类感兴趣，那么主要关心的是在不同可能的类 c 中，哪一个的 $p(C|\boldsymbol{E})$ 最大。因为本例中的 $\boldsymbol{E}$ 对所有类 c 都相同，所以我们可以只比较分子。

即使我们需要实际的概率估计，也同样可以避免计算分子中的 $p(\boldsymbol{E})$。这是因为类通常是互斥和穷尽的，即每个实例有且仅有一个类别值。在 Luxhote 的示例中，用户要么订房，要么不订。非正式地说，证据 $\boldsymbol{E}$ 要么属于 c_0，要么属于 c_1；用数学语言说，则是：

$$\begin{aligned} p(\boldsymbol{E}) &= p(\boldsymbol{E} \wedge c_0) + p(\boldsymbol{E} \wedge c_1) \\ &= p(\boldsymbol{E} \mid c_0) \cdot p(c_0) + p(\boldsymbol{E} \mid c_1) \cdot p(c_1) \end{aligned}$$

根据独立性假设，我们可以这样改写公式：

$$\begin{aligned} p(\boldsymbol{E}) = &\ p(e_1 \mid c_0) \cdot p(e_2 \mid c_0) \cdots p(e_k \mid c_0) \cdot p(c_0) \\ &+ p(e_1 \mid c_1) \cdot p(e_2 \mid c_1) \cdots p(e_k \mid c_1) \cdot p(c_1) \end{aligned}$$

将其与公式 9-3 结合，我们就得到了朴素贝叶斯方程，从而可以轻松地利用数据计算出后验概率：

注 5：顺便提一句，该假设比无条件独立的假设稍弱。

注 6：如果没有出现很多次，那么我们可以用小样本下的统计修正来计数。可参考 3.5 节。

$$p(c_0 \mid \boldsymbol{E}) = \frac{p(e_1 \mid c_0) \cdot p(e_2 \mid c_0) \cdots p(e_k \mid c_0) \cdot p(c_0)}{p(e_1 \mid c_0) \cdot p(e_2 \mid c_0) \cdots p(e_k \mid c_0) \cdot p(c_0) + p(e_1 \mid c_1) \cdot p(e_2 \mid c_1) \cdots p(e_k \mid c_1) \cdot p(c_1)}$$

虽然公式中存在许多项，但是每个项要么是某个单项证据的权重，要么是类先验概率。

9.3.2　朴素贝叶斯的优劣势

虽然朴素贝叶斯是个非常简单的分类器，但是它仍将所有特征证据都考虑在内，因而在存储空间和计算时间方面具有优势。模型训练仅包含存储所有实例的类的计数和特征的出现次数。如上所述，$p(c)$ 可以通过计算所有实例中 c 类实例的比例估计得知，$p(e_i|c)$ 则可以根据 c 类中含有特征 e_i 实例的比例估计得知。

虽然朴素贝叶斯非常“朴素”，且其独立性假设非常严格，但在许多现实分类问题上的表现却惊人地好。这是因为即使独立性假设被违反，分类器的性能也一般不会降低。试想两条强关联的证据。强关联意味着什么？大体上说，它意味着在看到其中一个时，也能看到另外一个。现在，如果把两者按相互独立处理，那么我们看到其中一个时就会说“存在某类别的证据”，而当看到另一个时则说“存在更多该类别的证据”。因此在某种程度上，我们会对证据重复计数。**然而**，只要证据的方向正确，那么重复计数就不会影响对分类的判断。事实上，它会导致概率估计在正确的方向上更为极端：概率会对正确的类做过高估计，而对错误的类（多个类）做过低估计。但分类时，我们选择的是概率最高的类，因此在正确方向上的极端估计并没有妨碍。

但如果我们要用概率估计值本身，这就成为了问题。因此如第 7 章所述，在实际进行成本收益的决策时，对朴素贝叶斯的使用必须谨慎。当概率的实际值与问题不相关时，业界人士的确会用朴素贝叶斯来排序，其各不同类别中仅包含实例的相对值。

朴素贝叶斯的另一个优势是，它是一种天然的“增量学习器”。增量学习器是一种能随训练更新模型的归纳技术。每出现一个新的训练实例它都会进行一次更新，且在出现新的训练数据时，它不需重新处理所有训练过的实例。

增量学习在应用过程中训练标签不断显露出来的情况下，尤其有优势。我们希望模型尽可能快地将这些新信息纳入模型内。比如，考虑创建个性化的垃圾邮件分类器的问题。当我收到垃圾邮件，我可以按一下浏览器中的“垃圾”按钮。这样不仅能把垃圾邮件从收件箱中删除，还能创建一个训练数据点：垃圾邮件的一个正样本个体。如果模型能立即更新、即时写入且立即把相似的邮件归为垃圾邮件，那么这个系统将非常有用。而朴素贝叶斯正是许多个性化的垃圾邮件监测系统的基础，比如 Mozilla Thunderbird 中的系统。

朴素贝叶斯几乎包含在所有数据挖掘工具包中，作为常见的基线分类器，它常常用于与更复杂的方法作比较。我们已经讨论了使用二元变量的朴素贝叶斯。上文呈现的这种基本思想可以轻松地扩展至多值类别型属性或数值型属性，你可以在数据挖掘算法的相关的教材中读到这些。

朴素贝叶斯的变体

确实有许多存在些许不同的分类器也叫朴素贝叶斯。这些区别往往很小，易被忽略(除了在这个补充栏里，本章其余篇幅都会忽略这些区别)。但它们会造成影响。

简而言之，朴素贝叶斯（NB）基于"生成"模型，即关于数据如何生成的模型。不同的NB基于不同的生成统计模型，而这些都构成了我们讨论过的主要NB假设（也就是，对每个类别而言，特征是条件独立地生成的）。此处我们虽然不会探讨实际的统计模型，但仍有必要考虑一个关键区别。

你会发现，我们描述的NB模型把每个特征值都当作支持或反对每个类的证据，可是如果存在许多特征呢，比如语言中的每个单词，或一个人可能会访问的每个网页？这种情况下，特征往往代表这些单词、网页等出现的次数或频率。事实证明，在这种应用场景下，大部分单词、网页等通常不会出现在任何具体实例中（如文件、线上用户）。

事实证明，朴素贝叶斯评分的计算有许多数学技巧，可以使我们只需要考虑现有的证据。感兴趣的读者不妨多读些与技巧相关的或与不同朴素贝叶斯模型相关的文献(McCallum & Nigam, 1998; Junqué de Fortuny 等, 2013)。结果是，这种大型稀疏域中的惯例是，仅对现有的证据做显性思考。因此，举个例子，在上文的广告示例中，我们通常只关注用户会**访问**的网站，而不关心用户**并未访问**的众多网站。后者将根据数据生成方式的假设，在数学中做隐性处理。同样，在下文中我们也将仅考虑Facebook用户点赞过的项目，而不会对用户没有点赞过的所有可能的项目做显性思考。

9.4 证据"提升度"的模型

8.5节展示了一种评估分类器的指标——**提升度**。提升度是正向类在选定的子总体中的比例与在整个总体中的比例之比。如果在随机选定的目标用户群中订房的概率是0.01%，而在我们选择的群体中概率是0.02%，那么分类器的提升度就是2，即我们选择的群体使订房率翻倍。

稍作调整后，我们便可以用朴素贝叶斯方程模拟由不同证据造成的不同提升度。这样的"稍作调整"指的是假设完全特征独立，而不是用于朴素贝叶斯的条件独立的弱假设。由于其对世界做了更强的简化假设，因而我们称其为"朴素朴素贝叶斯"。在假设完全特征独立后，公式9-3就变成了如下的朴素朴素贝叶斯：

$$p(c \mid \boldsymbol{E}) = \frac{p(e_1 \mid c) \cdot p(e_2 \mid c) \cdots p(e_k \mid c) \cdot p(c)}{p(e_1) \cdot p(e_2) \cdots p(e_k)}$$

可以在重新排列公式中的项后得到公式9-4。

公式9-4：作为提升度（lift）乘积的概率

$$p(c \mid \boldsymbol{E}) = p(c) \cdot \text{lift}_c(e_1) \cdot \text{lift}_c(e_c) \cdots$$

其中 $\text{lift}_c(\boldsymbol{x})$ 定义为：

$$\text{lift}_c(\boldsymbol{x}) = \frac{p(x \mid c)}{p(x)}$$

考虑这些提升度如何应用于新实例 $\boldsymbol{E} = <e_1, e_2, \cdots, e_k>$。从先验概率开始，每条证据（即每个特征 e_i）会根据一个等同于其提升度的因子（可能小于 1）提升或降低实例属于该类的概率。

概念上讲，我们先将一个数字（称为 z）设为类 c 的先验概率，然后观察示例，针对每个新证据 e_i，我们用 z 乘以其相应的提升度 $\text{lift}_c(e_i)$。如果提升度大于 1，概率 z 就会提升；如果小于 1，z 就会下降。

在 Luxhote 示例中，z 代表订房的概率，其初始值为 0.0001（看到证据前，某网站访问者订房的先验概率）。如果访问的是金融网站，就将订房概率乘以因数 2；如果是卡车拉力赛网站，就将订房概率乘以因数 0.25，以此类推。处理好 $\boldsymbol{E}$ 的所有证据 e_i 后，得到的乘积（称为 z_f）就是 $\boldsymbol{E}$ 属于类 c 的概率（信念）。而在本例中，网站访客 $\boldsymbol{E}$ 会预定房间。[7]

从这个角度考虑，你应该就能明白为何需要做独立性假设了——因为我们把所有证据按相互独立处理，所以只需用各自的提升度乘以 z。然而任何轻微的相依性都会导致终值 z_f 失真（它可能变得更高或更低）。因此，证据提升度及其组合形式非常有助于理解数据和**比较**实例的分数，但概率的实际终值必须审慎考虑。

9.5 示例：Facebook“点赞”的证据提升度

接下来本章基于真实数据来检验提升度。为了保持新鲜感，本章将换一个全新领域的应用问题。研究者 Michael Kosinski、David Stillwell 和 Thore Graepel 近期在《美国国家科学院院刊》上发表的一篇文章（Kosinski 等 , 2013）中展示了一些惊人的结果——社交网站 Facebook[8] 用户所“赞”的内容可以在很大程度上透露出通常并不明显的个人特征：

- 智力测验的水平
- 心理计量测验的水平（如开朗或尽责程度）
- 是否为（出柜了的）同性恋
- 是否饮酒或吸烟
- 宗教和政治观点
- 诸如此类

注 7：技术上讲，我们可能还需要考虑其没有访问其他网站的证据，该指标仍需要用一些数学技巧处理，详见前文的“朴素贝叶斯的变量”。

注 8：在此简单介绍一下 Facebook，以防你对它不太了解。Facebook 是一个可供大众分享各种关于兴趣和活动的信息以及联系“好友”的平台。Facebook 还有专门展示特殊兴趣的网页，如电视节目、电影、乐队、爱好等。和本章相关的是页面中“点赞”按钮。用户可以通过点击来表明自己对相应内容的喜爱。这样的“赞”通常可以被好友看到。并且，如果你“赞”过某个粉丝页，你就会逐渐看到许多与该粉丝页相关的推送。

读者不妨阅读此文，了解一下他们的实验设计。在读过本书后，你应该有能力理解大部分的结果。（比如，他们用 AUC 评估对二元特征的预测能力，此时你已经能够对此进行正确解读了。）

我们要做的是，寻找能为“高 IQ”提供强有力证据提升度的“赞”，或更具体地说，能为“在 IQ 测试中取得高分”提供强有力提升度的“赞”。我们要从 Facebook 用户群中选取一些样本，并定义二元目标变量“IQ > 130”。

接下来，我们来找一找提供最高提升度的“赞”……[9]（结果见表 9-1）

表9-1：一些Facebook页面的“赞”及相关提升度

点赞的页面	提　升　度	点赞的页面	提　升　度
《指环王》	1.69	维基解密	1.59
一漫画	1.57	贝多芬	1.52
科学	1.49	美国国家公共电台	1.48
心理学	1.46	《千与千寻》	1.45
《生活大爆炸》	1.43	跑步	1.41
Paulo Coelho	1.41	Roger Federer	1.40
《每日秀》	1.40	《星际迷航》（电影）	1.39
《迷失》	1.39	哲学	1.38
《别对我说谎》	1.37	《洋葱报》	1.37
《老爸老妈的浪漫史》	1.35	《科尔伯特报告》	1.35
《神秘博士》	1.34	《星际迷航》	1.32
《哈尔的移动城堡》	1.31	Sheldon Cooper	1.30
《电子世界争霸战》	1.28	《搏击俱乐部》	1.26
愤怒的小鸟	1.25	《盗梦空间》	1.25
《教父》	1.23	《单身毒妈》	1.22

那么，根据上文中的公式 9-4 和当时所做的独立性假设，我们便可以基于某人点赞过的页面，计算此人 IQ 极高的概率。Facebook 上，赞过 Sheldon Cooper 页面的用户的高 IQ 概率比一般人群的高 IQ 概率高出 30%，而赞过《指环王》页面的用户的高 IQ 概率比一般人群的高 IQ 概率高出 69%。

当然，有一些页面上的赞也会**拉低**用户**高 IQ** 的概率。但我们不会在这里列出这些页面，以防你因此感到沮丧。

本例还说明了根据数据收集过程，谨慎考虑结果含义的重要性。上文的结果并非表示喜欢《指环王》代表用户很可能具有高 IQ，而是表示赞过《**指环王**》的 Facebook 页面代表用户很可能具有高 IQ。两者的区别非常重要：在网页上点“赞”并不等同于喜欢这个内容，而我们收集到的数据是前者，并非后者。

注 9：感谢 Wally Wang 在生成结果方面的鼎力相助。

9.6 小结

前一章所展示的建模技术，主要提出了在实例总体的不同分组中“区分目标变量值的最佳方法是什么”的问题。分类树和线性方程都以尝试降低损失或熵值的方式构建模型，而损失和熵值都是区分度的函数。分类树和线性方程都被称为**判别分类**法，因为它们能直接辨别不同的目标变量。

本章则介绍了一类新方法，它能够在本质上将该问题颠倒为：“不同的目标分组如何**生成**特征值？”这些方法能够模拟数据生成的过程，而在使用环节，出现需要分类的新实例时，它们就会用模型来回答这个问题：“哪个类最有可能生成这个实例？”因此，该方法在数据科学中被称为**生成方法**。这其中一大类常用方法被称为**贝叶斯方法**。它们之所以是生成方法，是因为它们严格依赖于贝叶斯法则。贝叶斯方法的相关文献博大精深，你在数据科学领域也会发现它们非常常见。

本章首先集中讨论了一种尤其常见、尤其简单，但行之有效的贝叶斯方法——朴素贝叶斯分类器。它的“朴素”在于，由于模型中的特征（根据每个目标）被当作是独立生成的，因而当特征实际上相关时，分类器最后可能会重复计算证据数。由于简单，朴素贝叶斯非常快速高效，而且它虽然“朴素”，但是却惊人地有效。由于简单，它甚至成为了数据科学中常用的“基线”方法（任何新问题都会首先使用的方法）之一。

本章还探讨了使用某种独立性假设的贝叶斯推理如何帮助通过我们计算“证据提升度”检验大量可能的证据是否支持结论。本章还举了一个例子：“赞”过《搏击俱乐部》《星际迷航》或 Sheldon Cooper 的 Facebook 页面的用户拥有高 IQ 的概率比一般人群高 30%。

第10章

文本的表示和挖掘

基本概念：构造挖掘友好型数据表示法的重要性；数据挖掘所用文本的表示

示例方法：词袋模型表示法；TFIDF 计算；n-grams；词干提取；命名实体提取；主题模型

到目前为止，本书一直在忽略或回避数据挖掘流程的一个重要环节——数据准备。大部分数据挖掘方法是以特征向量为输入的，然而在现实中，我们获得的数据并非都是以特征向量形式表示的。数据总是以它们在问题中自然产生的方式呈现，如果我们想运用手头上的诸多数据挖掘工具，就必须把数据加工处理成为适合工具的表示方式，或者构造适合数据的新工具，而一流的数据科学家会同时采用两种策略。通常，首先用现有工具对数据进行处理会比较简单，因为现有工具不但比较好理解，而且种类很多。

本章将关注一种特别的数据类型：文本数据。如今，由于互联网已成为无处不在的沟通渠道，文本数据变得极为常见。通过检验文本数据，我们可以看到数据工程中许多真正的复杂性，并且能加深对一种非常重要的数据的理解。到第 14 章你会明白，虽然本章仅关注文本数据，但这些基本原则确实能推广到其他重要的数据类型中。

我们在 6.4.4 节中遇到过一次文本数据。当时我们有意回避了对新闻报道数据准备过程的详细探讨，因为当时关注的是聚类，而文本的准备有些偏题。本章则专门讨论文本处理的难点和机会。

原则上，文本不过是数据的另一种形式，文本处理也只是表示工程的特殊情形。实际上，处理文本不仅需要专用的预处理步骤，有时还需要数据科学团队具有特定的专业知识。

关于文本挖掘，有各种图书、会议、公司专门对其进行研究和讨论。然而本章只是浅尝辄止，对技术和典型商业应用中的问题进行概述。

首先讨论文本的重要性和它难以处理的原因。

10.1 为什么文本很重要

文本无处不在。许多传统应用程序仍会产生或记录文本。病历、用户投诉记录、产品查询记录和维修记录仍是人与人（而非计算机）之间的主要交流方式，因此仍需将其“编码”为文本。要想对这类庞大的数据进行开发利用，必须将其转换成有意义的形式。

互联网也许是培育“新媒体”的温床，但它的大部分仍与旧媒体形式相同，包含大量个人网页、Twitter 简讯、电子邮件、Facebook 状态更新、产品介绍、Reddit 评论和博文等形式的文本。我们每天使用的搜索引擎（谷歌和必应）就是由大量面向文本的数据科学支撑的。虽然音乐和视频占据了大部分的网络流量，但人们线上交流的主要方式还是文本。确实，Web 2.0 的主旨是，网站不仅能为用户提供以社区形式交流的平台，还能让用户生成更丰富的网页内容，而用户生成的内容和交互通常为文本形式。

在各个行业中，理解用户的反馈通常需要理解文本，但情况并非总是如此。不可否认，一些重要的用户态度可以用数据明确地表示，或可以通过行为推断，比如五星级评定、点击模式、转化率等。我们也可以花点钱，用焦点小组和线上调查等方法来收集和量化数据。但在许多情况下，如果想“聆听用户的意见”，那么就得亲自去读此人所写的内容，如产品评论、客户反馈表格、意见书、电子邮件等。

10.2 为什么文本很难处理

文本往往被称作“非结构化”数据。这指的是文本中不含一般数据所具备的结构：由有固定意义的域构成的记录表格（也就是特征向量的集合），以及表格之间的关联关系。虽然文本中的确存在大量结构，但是这些结构是**语言学**结构，它们可供人类理解，但计算机无法理解。

单词的长度和文本域中所含的单词数都会不同。有时单词的顺序会影响含义，有时又不会。

从数据角度看，文本相对较**脏**，因为人们写东西常常不合文法，总是犯拼写错误、把词连在一起、胡乱缩写和乱加标点。即使文本的表达完美无瑕，其中也可能存在同义词（多词同义）和同形异义词（一词多义）。一个领域中的术语和简写对另一个领域而言可能毫无意义，比如我们不能强求医疗记录和计算机维修记录包含相同的术语。最坏的可能是，两者的术语含义甚至存在冲突。

由于文本的目的是方便人们之间的交流，故而**语境**非常重要，甚至比在其他数据格式中更重要。思考以下的影评片段：

> “电影的第一部分远好过第二部分。演技很差，到最后甚至失控了。暴力部分过头了，而结尾也令人难以置信。但这仍不失为一部有趣的电影。”

整段话到底是褒还是贬呢？**难以置信**一词是褒义还是贬义？在不考虑整个语境的情况下，评估任何单词或短语都是很难的。

因此，文本在输入数据挖掘算法前，必须经过大量的预处理。通常，文本的特性越复杂，文本问题所包含的方面就越多。本章接下来将仅描述准备数据挖掘所用文本的一些基本方法。

10.3　表示法

探讨完文本的棘手之处后，我们来看看将文本的正文转化成能直接输入数据挖掘算法的数据集的基本步骤。文本挖掘的一般策略是，在所有可用的技术里选择最简单的（也就是最便宜的）。虽然如此，但是这些概念却是许多网页搜索引擎（如谷歌和必应）背后的关键技术。接下来的一个例子将演示基本查询检索。

首先介绍一些基本术语。这些术语大部分源自信息检索（IR）领域。**文档**指一段文本，无所谓长短。它既可以是一个句子，也可以是 100 页的报告，还可以介于两者之间，如一条 YouTube 评论或一篇博文。一般说来，一篇文档中的所有文本会被放在一起加以考虑，并在匹配或分类时，将所有文本作为单独一项进行检索。文档由单独的**语符**（token）或**词语**（term）构成。目前你可以暂时将语符或词语视作单词。随着学习的深入，你会知道它们与我们平时谈论的单词的区别。文档的集合则被称为**语料库**（corpus）。[1]

10.3.1　词袋模型

请谨记文本表示任务的目的。本质上，我们把一组文档（每一篇都是形式自由的单词序列）转化为熟悉的特征向量形式。每篇文档都是一个数据项，而我们事先不知道它们的特征是什么。

首先要介绍的方法叫作“词袋模型”。顾名思义，该方法把每篇文档作为单词的集合，忽略语法、词序、句型结构和标点。它把文档中的每个单词都作为可能的重要关键词。该表示法非常简单，生成成本不高，且适用于许多任务。



集合和包

尽管**集合**和**包**在数学中有特殊含义，但都不是这里所指的含义。集合中每个项只能出现一次，而我们却想要考虑单词的出现次数。**包**在数学中指的是**多重集**，即其中的成员可以出现不止一次。词袋表示法首先把文档当作单词的包（即多重集），而忽略词序及其他语言结构。然而，用于文本挖掘的表示法通常比单纯计算词频更加复杂，下文会介绍。

那么，如果每个单词都有可能是特征，那么文档的特征值又是什么？有很多对应方法，其中最基本的方法将每个单词视作一个语符，并把每篇文档用 1（文档中存在该语符）或 0（文档中不存在该语符）表示。该方法将文档简化为其中所包含的一组单词。

10.3.2　词频

下一步是用文档中的字数（词频）代替 0 或 1，这能区分单词使用的次数。在某些应用场景中，词语的重要性应随其在文档中出现的次数增多而提升。这就叫作**词频表示法**。请思考表 10-1 中三个非常简单的句子（文档）。

注 1：body 的拉丁文。复数形式为 corpora。

表10-1：三篇简单文档

d1	jazz music has a swing rhythm
d2	swing is hard to explain
d3	swing rhythm is a natural rhythm

每个句子被视为一篇独立的文档。用词袋法对其词频进行整理后，会形成如表 10-2 的表格。

表10-2：词语计数表示法

	a	explain	hard	has	is	jazz	music	natural	rhythm	swing	to
d1	1	0	0	1	0	1	1	0	1	1	0
d2	0	1	1	0	1	0	0	0	0	1	1
d3	1	0	0	0	1	0	0	1	2	1	0

一般在把单词写入表格前，要做一些基本处理。再思考以下更复杂的样本文档：

> Microsoft Corp and Skype Global today announced that they have entered into a definitive agreement under which Microsoft will acquire Skype, the leading Internet communications company, for $8.5 billion in cash from the investor group led by Silver Lake. The agreement has been approved by the boards of directors of both Microsoft and Skype.

表 10-3 把文档简化为词频表示法的形式。

表10-3：经过标准化和词干提取后的词语，按频率排序

词语	计数	词语	计数	词语	计数	词语	计数
skype	3	microsoft	3	agreement	2	global	1
approv	1	announc	1	acquir	1	lead	1
definit	1	lake	1	communic	1	internet	1
board	1	led	1	director	1	corp	1
compani	1	investor	1	silver	1	billion	1

为了将样本文档转化为上述表格，我们执行了以下步骤。

- 首先，统一字母的大小写，将每个单词都变为小写，从而使 Skype 和 SKYPE 相同。因为由大小写不同而产生的单词变体非常常见（比如 iPhone、iphone 和 IPHONE），所以统一大小写一般非常必要。
- 然后，对一些单词进行**词干提取**，去除它们的后缀，使类似于 announces、announced 和 announcing 的这样动词全都转化为 announc。同样，名词的复数形式也要转化为单数形式，因此文中的 directors 在表格中变为了 director。
- 最后，删除**停用词**。停用词是在英语（或任何一种需要解析的语言）中极其常见的词，比如 the、and、of 和 on，一般需要删除。

注意，文中的“$85”被删掉了。是否应该如此呢？尽管数字通常被视作文本处理中不重要的细节，但这应根据表示法的目的来决定。你可以想想，“4TB”和“1Q13”之类的术语在哪些语境下毫无意义，在哪些语境下又至关重要。



随意删除停用词

提醒一句：停用词并不总是需要删除，比如，这些词在标题中就至关重要。像 Cormac McCarthy 的 *The Road*（一对父子在世界末日后求生的故事）就与 John Kerouac 的著名小说 *On the Road* 大相径庭，而不加考虑地直接删除停用词将导致两者没有区别。同样，最近上映的惊悚片 *Stoker* 也不应与 1935 年的喜剧电影 *The Stoker* 混淆。[2]

表 10-3 展示了词语的原始计数。但是一些系统不会使用原始计数，而是会根据文档长度，对词频进行标准化。使用词频的目的是表示词语与文档的相关性。因为长文档的单词往往比短文档多，所以单词的出现次数也更多。但这并不意味着长文档比短文档更重要，或相关性更强。为了根据文档长度进行校正，需要用一些方法对原始词频进行标准化，比如将其除以文档的总词数。

10.3.3 度量稀疏度：逆文档频率

既然词**频**度量的是一个词语在一篇文档中的普遍程度，那么在决定词语的权重时，我们可能还想知道该词在整个语料库中的普遍程度。这个问题有两种相反的思考方式。

首先，一个词语不能太**罕见**。如果一个不常见的单词 pr☺ensile 仅在语料库的一篇文档里出现过，那么这个词重要吗？这要视应用情景而定。在检索时，这个词可能很重要，因为用户寻找的是这个确切的词；而在分类时，却没必要保留一个只出现过一次的词，因为它绝对不可能成为一个有意义的簇的构成依据。因而，文本处理系统通常会给某词语必须在其中出现的文档数设定一个较小的（任意的）下限。

其次，从相反角度考虑，一个词语也不能太**常见**。在每篇文档里都出现的词语不但对分类没有帮助（它分辨不出什么来），而且也不会是簇的构成依据（不然整个语料库都会聚在一起）。

过于常用的词语通常会被删掉，而实现方法之一是给可出现词语的文档数（或文档所占比例）设定一个任意的上限。

除了给词频设定上下限，许多系统还会考虑词语在语料库中的分布。包含一个词语的文档越少，则在这些文档中，该词语的重要性就可能越高。词语 t 的稀疏度一般用**逆文档频率**（IDF）来度量，如公式 10-1 所示。

公式 10-1：词语的逆文档频率

$$\text{IDF}(t) = 1 + \log\left(\frac{\text{文档总数}}{\text{包含}t\text{的文档数}}\right)$$

一个词语越罕见，其 IDF 就越高。图 10-1 中，语料库共含 100 篇文档，而 IDF(t) 为 t 出现过的文档数目的函数。如你所见，当词语非常罕见时（在图像的最左侧），IDF 极高。而

注 2：这些例子都来自一款流行的搜索引擎近期对影评网站的搜索结果。不是每个人都会注意停用词删除的问题。

随着 t 在文档中越来越常见，IDF 会快速下降，最终渐近于 1.0。由于大部分停用词非常常见，因而其 IDF 通常接近 1。

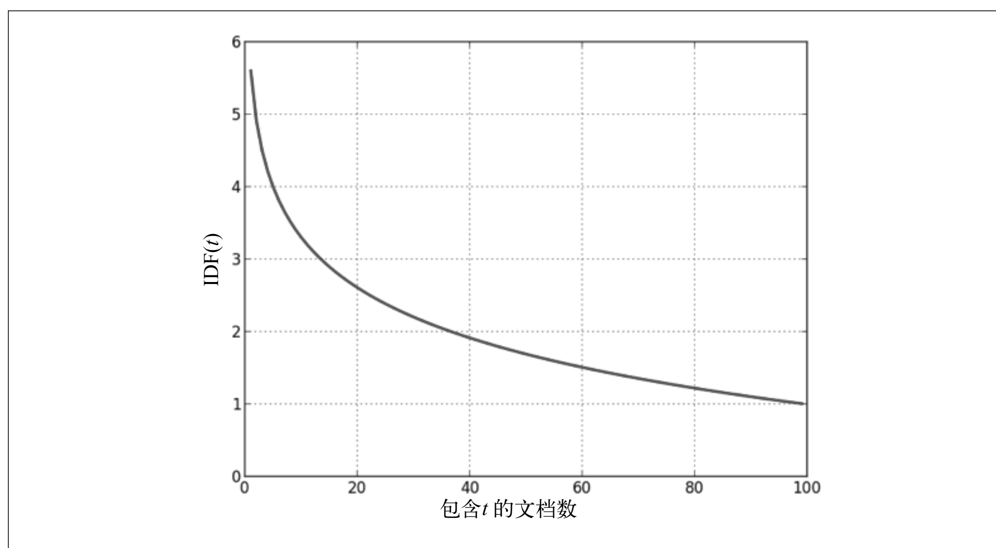


图 10-1：100 篇文档的语料库中，词语 t 的 IDF

10.3.4 TFIDF

有一种非常流行的文本表示法是词频（TF）和逆文档频率（IDF）相结合的产物，俗称 TFIDF。给定文档 d，词语 t 的 TFIDF 值的计算方法是：

$$\text{TFIDF}(t, d) = \text{TF}(t, d) \times \text{IDF}(t)$$

注意，TFIDF 针对的是单篇文档（d），而 IDF 则依赖整个语料库。使用词袋表示法的系统在进行词语计数前，通常需要提取词干和删除停用词。由文档中的词语计数得到每个词语的 TF 值，由语料库中的文档计数得到 IDF 值。

因此，每篇文档都变成了一个特征向量，而语料库则是这些特征向量的集合，可用于数据挖掘中的分类算法、聚类算法或检索。

因为文档中含有许多可能具有文本表示功能的词语，所以经常要用到特征选择。特征选择的方法有很多，比如给词语计数附加最小阈值或最大阈值，或按照诸如信息增益（详见 3.2.2 节）这样的指标来将词语按重要程度排序，从而剔除信息增益较低的词语。

词袋文本表示法把文档中的每个词作为独立的潜在关键词（特征），然后基于词频和罕见程度给每篇文档赋值。虽然 TFIDF 是一种常见的词语值表示法，但它未必是最优的方法。如果有人用词袋表示法描述对语料库的挖掘，那么这意味着他把每个词视为独立的特征。它们可以是二值型、词频或 TFIDF，可以标准化，也可以不标准化。虽然数据科学家已经培养出了直觉，可以找到解决给定文本问题的最佳方案，但他们往往会首先试验不同的表示法，看看哪个的结果最佳。

10.4　示例：爵士音乐家

在介绍过一些基本概念后，我们来用一个表示爵士音乐家的具体示例来说明它们。本例中，我们将思考一个包含 15 位杰出爵士音乐家及其维基百科个人简介片段的小型语料库。以下是几位音乐家个人简介的片段。

- Charlie Parker

  Charles“Charlie”Parker Jr.，美国爵士萨克斯管演奏家、作曲家。Miles Davis 曾说："爵士乐历史用四个词就可以概括：Louis Armstrong、Charlie Parker。"Parker 早期有着"新兵"的昵称，简称为"兵"，这个昵称一直跟着他，并且多次激发了 Parker 作曲的灵感，[……]

- Duke Ellington

  Edward Kennedy“Duke”Ellington，美国作曲家、钢琴家、大型爵士乐队指挥，作曲超过 1000 首。《波士顿环球报》的 Bob Blumenthal 如是评价他："在 Edward Kennedy Ellington 出生以后的一个世纪中，无论美国还是其他国家，都没有比他更伟大的作曲家。"尽管 Ellington 是爵士历史中的一名重要人物，但其涉足的音乐流派甚广，包含布鲁斯、福音、电影配乐、流行音乐和古典音乐等。[……]

- Miles Davis

  Miles Dewey Davis III，美国爵士音乐家、小号手、乐队指挥、作曲家，被公认为 20 世纪最具影响力的音乐家之一。Miles Davis 与其乐队处于爵士音乐许多重大发展的前沿，如比波普爵士乐、冷爵士乐、硬波普乐、调式爵士乐和融合爵士乐。[……]

尽管该语料库只有 15 篇文档，但整个语料库及其词汇却庞大到无法在这里全部展示（提取词干和删除停用词后仍有将近 2000 个特征），因此我们只用一个样本来说明。思考以下句子："Famous jazz saxophonist born in Kansas who played bebop and latin."（生于堪萨斯的知名爵士萨克斯管演奏家，演奏比波普爵士乐和拉丁。）如果把这句话输入搜索引擎，它将如何表示？答案是，它将被作为文档处理，也会经历许多同样的步骤。

首先，进行基本的词干提取。虽然词干提取的方法并非万无一失，有可能把 Kansas 和 famous 转化为 kansa 和 famou（两个词均无意义），但只要全文保持一致，这种错误就无伤大雅。词干提取的结果如图 10-2 所示。

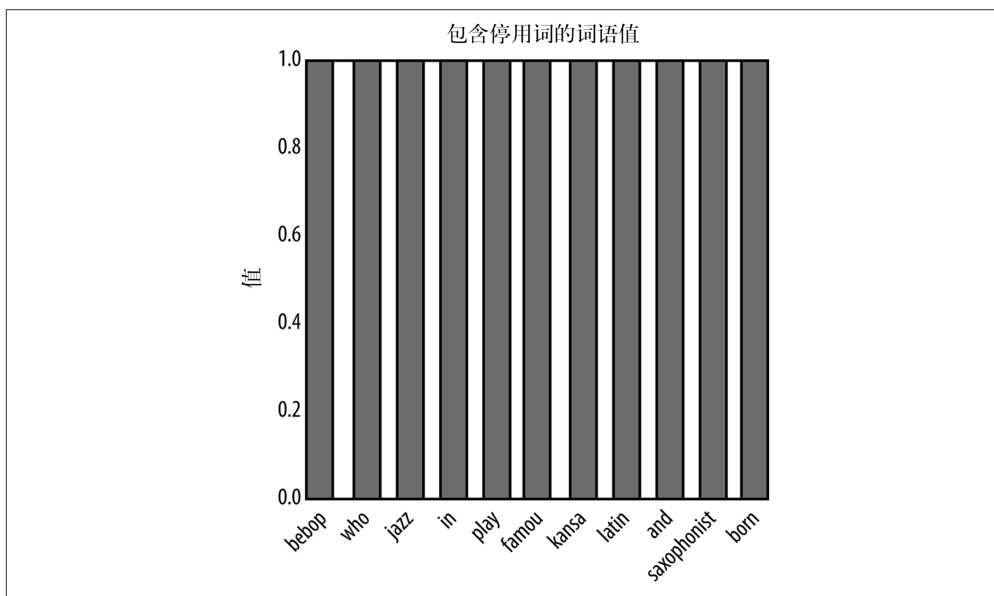


图 10-2：词干提取后，查询语句“Famous jazz saxophonist born in Kansas who played bebop and latin”的表示

接下来，删除停用词（in 和 and），并将单词根据文档长度标准化。结果如图 10-3 所示。

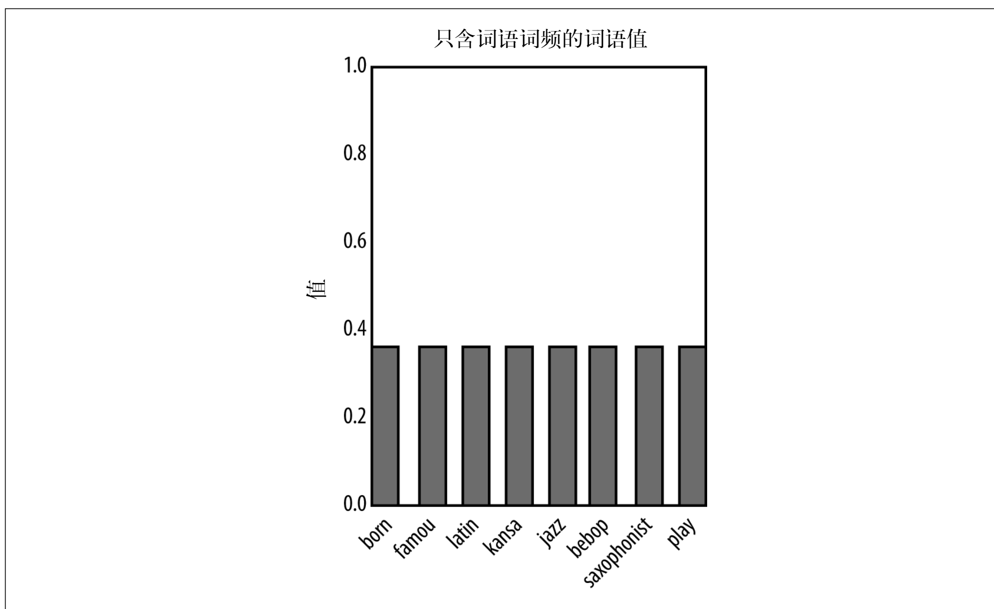


图 10-3：删除停用词并对词频进行标准化后，查询语句“Famous jazz saxophonist born in Kansas who played bebop and latin”的表示

如果到此为止，那么这些值通常会被用作词频（TF）的特征值。但是此处我们会通过把每个词语的 TF 值与 IDF 值相乘，得到完整的 TFIDF 表示。如我们所说，该指标会提升罕见单词的权重。

因为 jazz 和 play 在爵士音乐家个人简介语料库中出现得非常频繁，几乎可以视作停用词，所以其权重没有得到 IDF 的提升。

因为 TFIDF 值最高的词语（latin、famous 和 kansas）是语料库中最罕见的词，所以它们在查询语句中的权重最高。最后，重新对词语进行标准化，得到最终的 TFIDF 权重，如图 10-4 所示。这就是样本“文档”（查询语句）的特征向量表示。

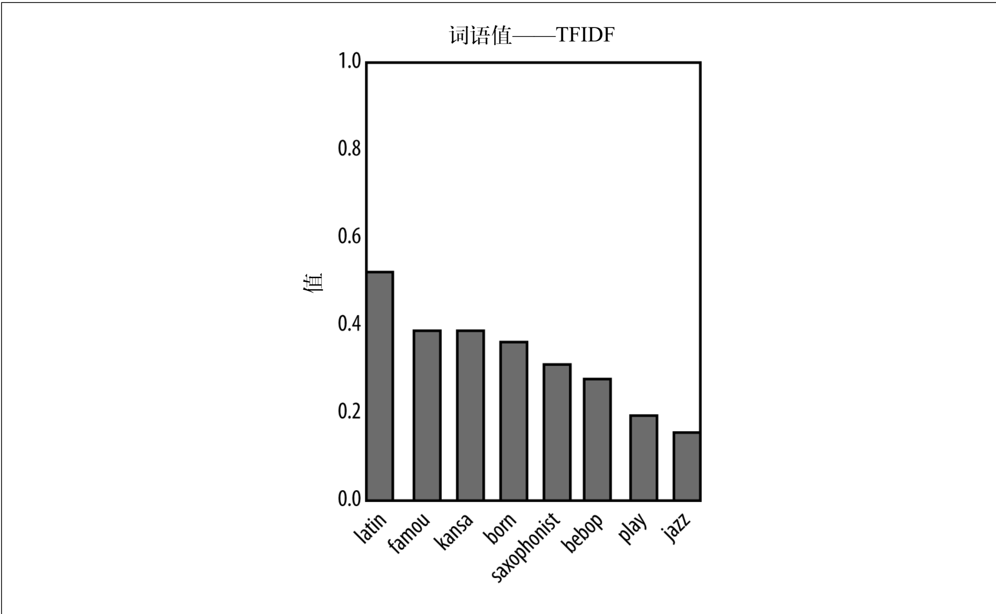


图 10-4：查询“Famous jazz saxophonist born in Kansas who played bebop and latin”的最终 TFIDF 表示

看过这篇小“文档”的表示形式后，我们来用它做点什么。还记得第 6 章中探讨的用距离测度进行最近邻检索吗？当时我们展示了一个检索相似威士忌的例子，现在也可以这么做。如果把样本句子“Famous jazz saxophonist born in Kansas who played bebop and latin”输入简易的搜索引擎，该引擎会如何运作？首先，它会把这句查询转化为 TFIDF 表示，如图 10-4 所示。我们已经计算过了每篇爵士音乐家个人简介的 TFIDF 表示，现在只需要再计算出这句查询与每篇个人简介的相似度，然后选择最相似的就可以了！

为此，我们选用 6.3.2 节中探讨过的余弦相似性函数（见公式 6-5）。余弦相似性在文本分类中常用于度量文档之间的距离。

如表 10-4 所示，与查询文档最匹配的爵士音乐家是 Charlie Parker，这个人的确是一位生于堪萨斯的萨克斯管演奏家，他演奏比波普爵士乐，有时也会结合其他音乐流派，包括拉丁。这些在他的简介中都有所提及。

表10-4：每篇音乐家文本与查询“Famous jazz saxophonist born in Kansas who played bebop and latin”的相似度，按相似度降序排序

音乐家	相似度	音乐家	相似度
Charlie Parker	0.135	Count Basie	0.119
Dizzie Gillespie	0.086	John Coltrane	0.079
Art Tatum	0.050	Miles Davis	0.050
Clark Terry	0.047	Sun Ra	0.030
Dave Brubeck	0.027	Nina Simone	0.026
Thelonius Monk	0.025	Fats Waller	0.020
Charles Mingus	0.019	Duke Ellington	0.017
Benny Goodman	0.016	Louis Armstrong	0.012

10.5 *IDF和熵的关系



前方有技术细节！

在刚开始讨论预测建模时，3.2.1 节介绍了熵测度，感兴趣（且记忆力强）的读者可能发现了，逆文档频率和熵有些相似，两者似乎都能度量一个集合中属性的“混合”程度。两者之间是否有联系？它们是不是相同的概念？答案是，它们虽然并不相同，但的确相关。本节将展示两者的联系，如果对此不感兴趣，你可以跳过本节。

图 10-5 展示了一些与我们将要探讨的公式相关的图像。首先，假设 t 是文档集中的一个词语，那么 t 在文档集中出现的概率是多少？我们可以这样估计：

$$p(t)=\frac{\text{包含}t\text{的文档数}}{\text{文档总数}}$$

为简化问题，自此我们将用 p 替代估计值 $p(t)$。回忆一下，词语 t 的 IDF 值的定义是：

$$\text{IDF}(t)=1+\log\left(\frac{\text{文档总数}}{\text{包含}t\text{的文档数}}\right)$$

1 是个常数，可以直接忽略，然后你会发现，IDF(t) 其实就是 $\log(1/p)$，而在代数学中，$\log(1/p)$ 等于 $-\log(p)$。

再次思考含有词语 t 的文档集，其中每篇文档要么含有 t（概率为 p），要么不含（概率为 $1-p$）。我们用一个伪镜像词语 *not_t* 表示文档中不含 t 的概率。那么该词语的 IDF 值是多少呢？如下：

$$\text{IDF}(not_t)=\log 1/(1-p)=-\log(1-p)$$

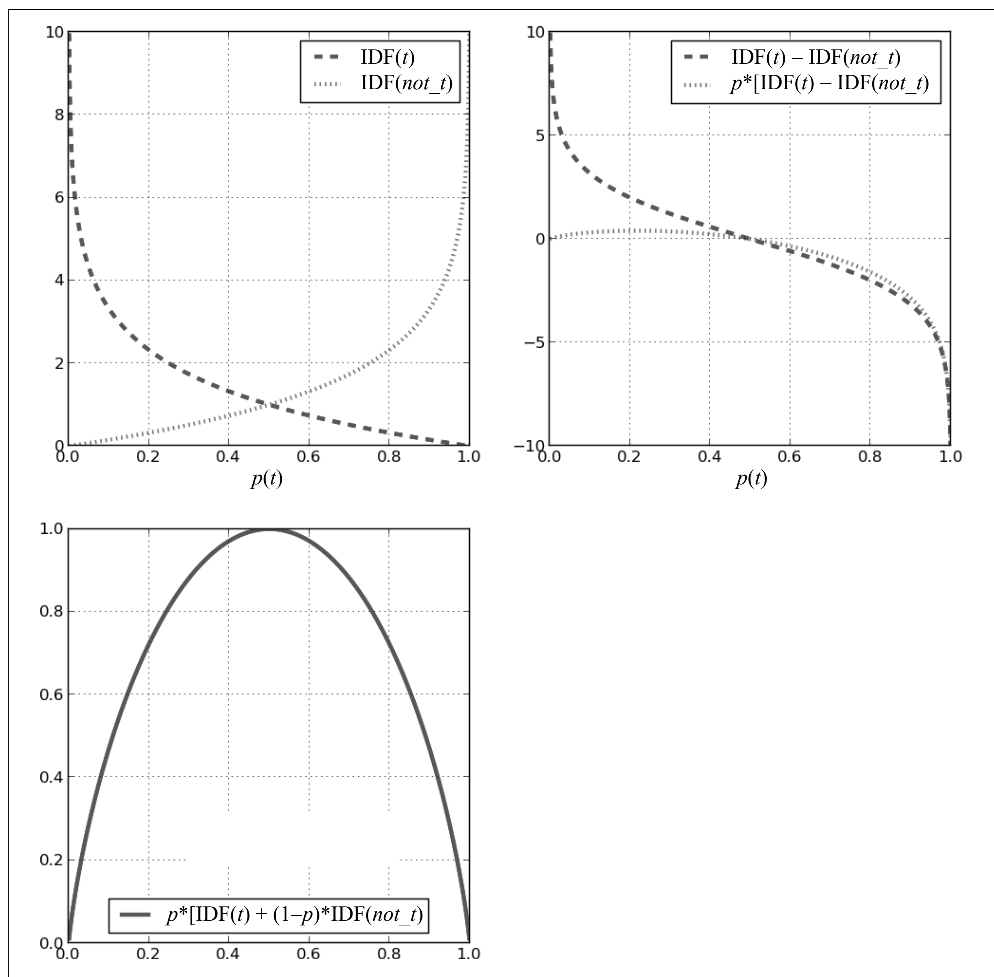


图 10-5：与 IDF(*t*) 和 IDF(*not_t*) 相关的不同值的多幅图像

先看图 10-5 中左上角的图像，正如我们所料，其中两条图线互为镜像。然后再回忆公式 3-1 对熵的定义，对于一个 $p_2 = 1 - p_1$ 的二值型词语而言，其熵值为：

$$熵 = - p_1 \log (p_1) - p_2 \log (p_2)$$

本例中，二值型词语 t 要么出现在文档中（概率为 p），要么不出现（概率为 $1 - p$），因而根据 t 划分的文档集的熵的定义式可简化为：

$$熵(t) = - p \log (p) - (1 - p) \log (1 - p)$$

现在，根据 IDF(*t*) 和 IDF(*not_t*) 的定义，我们可以开始替换和简化了（可参考图 10-5，右上角的图包含许多这样的子表达式）。

$$
\begin{aligned}
\text{熵}(t) &= -p\log(p)-(1-p)\log(1-p) \\
&= p\cdot \text{IDF}(t)-(1-p)[-\text{IDF}(not_t)] \\
&= p\cdot \text{IDF}(t)+(1-p)[\text{IDF}(not_t)]
\end{aligned}
$$

你会发现，现在这个式子是计算**期望值**的形式了！我们可以根据 *t* 在语料库中出现的概率，将熵值表示成 IDF(*t*) 和 IDF(*not_t*) 的期望值的形式。图 10-5 左下角的图也的确和图 3-3 中的熵值曲线相符。

10.6 词袋模型之外的方法

基本的词袋模型方法相对简单，却有许多可取之处。它不需要复杂的解析能力和其他语言学分析，却在许多工作中表现惊艳，因而往往是数据科学家解决新文本挖掘问题时的首选。

但仍有一些应用场景是词袋模型不太适用的，此时就需要采用更复杂的技术。本节将简要介绍其中几种。

10.6.1 n-grams序列

如前文所示，词袋表示法将每个单词作为一个词语，完全忽略词序。但有时候，词序也很重要，其信息需要在表示中保留。增加复杂度的下一步就是把相邻的单词**序列**也视作词语，比如可以将相邻的两个单词视为词语，这样一来，文档中包含的一句“The quick brown fox jumps”就可以变为一个集合，包含单词 {quick, brown, fox, jumps}，加上表征“quick_brown”“brown_fox”和“fox_jumps”。

这种通用表示手法叫作 **n-grams**。相邻的两个单词通常叫作 2-grams。如果一名数据科学家提到把文本表示为“最大为 3 的 n-grams 词袋”，他指的是把每篇文档中的单个单词、相邻两个单词和相邻三个单词组作为文档特征对文档进行表示。

n-grams 适用于特定词组比较重要，而组成词组的单词却意义不大的情况。比如在商业新闻中，3-gram 的“exceed_analyst_expectation”就比分别出现的 analyst、expectation 和 exceed 有意义得多。n-grams 的优势是容易生成，不要求使用者掌握语言学知识或复杂的解析算法。

n-grams 的主要劣势是其极大地扩大了特征集。由于文档中存在许多相邻的两个单词和许多相邻的三个单词，因而所产生的特征的数量会迅速增加。而且，其中许多单词组非常罕见，可能只在语料库中出现过一次。如果要在数据挖掘中应用 n-grams，就必须额外考虑处理大量特征的问题（比如特征选择）和计算存储空间的问题。

10.6.2 命名实体提取

有时我们还需要继续提升短语提取的复杂度。我们需要识别文档中的常见命名实体。Silicon Valley、New York Mets、Department of the Interior 和 Game of Thrones 都是重要的短语，虽然这些短语中的单词也可能有意义，但并不重要，而它们在生成独一无二的命名实

体后，就拥有了有趣的特性。基础的词袋表示法（甚至 n-grams）并不能捕获这些有趣特性，因而我们想用一个预处理组件来得知何时单词序列中包含合适的名称。

许多文本处理工具包都包含某种命名实体提取器，它们通常可以处理原始文本，并提取出被标注为人名或组织名的短语。举个例子，有时在经过标准化后，HP、H-P 和 Hewlett-Packard 等短语都与惠普公司的常见表示有关。

词袋模型和 n-grams 都按照空格和标点对文本进行划分，而命名实体提取器则属于知识密集型。为了取得比较好的效果，必须先在大型语料库上训练它，或手动为其录入大量命名信息。没有语言学原则规定“奥克兰突袭者”一定指代那支职业足球队，而非一群加利福尼亚激进投资者。这样的知识需要经过学习，或被手动录入。实体识别的特性各不相同，有的提取器专门针对某个特定专业领域，如工业、政府和流行文化等。

10.6.3　主题模型

我们已经学习了直接根据文档中的单词（或命名实体）构建的模型，该模型（不管最后如何）直接涉及单词。这样直接的模型虽然学习起来相对高效，但并非总是最优的选择。由于语言和文档的复杂性，有时我们想在文档和模型之间额外加入一层，在关于文本的语境下，我们称这层为**主题层**。

主题层的中心思想是，首先对语料库的主题集合分别建模。像之前一样，我们把每篇文档视作一个单词序列，但是这次不直接把单词用于最后的分类器，而把单词映射到一或多个主题中。这些主题同样需要从数据中学习（通常是通过无监督的数据挖掘）。而最终的分类器则依据中间的主题来定义，而非单词。设定主题层的一个优势是（比如在搜索引擎中）查询可以使用与某文档中特定单词并不完全匹配的词语。只要所查询的词语映射到了正确的主题（可以是多个主题），该文档就仍然可被认为与该查询相关。

构建主题模型的一般方法包括矩阵因子分解方法（如潜在语义索引）和概率主题模型（如隐含狄利克雷分配）。这些方法中的数学知识超出了本书的范围，但你可以把主题层想象成单词的聚类。在主题建模中，词语与主题相关联，词语权重则通过主题建模过程**学习**。与聚类相同，主题从数据的统计规律性中显现。同样，这些主题既不一定容易理解，也不一定为我们所熟知（尽管在很多情况下它们是这样的）。



主题是隐含信息

主题模型是一种**隐含信息**模型，第 12 章将（与电影推荐示例一起）进一步探讨它。隐含信息可以理解为信息中一种未被观测到的中间层，处于输入层与输出层之间。寻找文本中的隐含主题和寻找观影者的隐含“品味”维度这两种技术在本质上是相同的。在文本中，不仅要将单词映射到（未被观测到的）主题，还要将主题映射到文档，而这虽然使得整个模型更加复杂、学习成本更高，但也会使其性能更好。另外，隐含信息本身往往就很有趣、很有用（你将在第 12 章的电影推荐示例中再次看到）。

10.7 示例：通过挖掘新闻报道预测股价变动

为阐述文本挖掘中的一些问题，我们将引入一个新的预测挖掘任务：根据新闻报道的文本预测股价波动。大体上说，我们要根据新闻报道预测股票市场。这个任务包含了许多关于文本处理和问题界定的通用元素。

10.7.1 任务

股票市场在每个交易日都会有所变动，企业会进行决策并宣布决策，如并购、发布新产品、收益预期等，而金融新闻行业会对此进行报道。在读过这些新闻报道后，投资者可能会改变对报道中所提及公司的前景的预期，因而交易股票，导致股价变化。举个例子，收购、收益、监管制度变化之类的公告之所以会影响股价，是因为它们或是直接影响了潜在的收益，或是影响了交易者对其他交易者收购价格的判断。

当然，以上针对金融市场的观点是高度简化的。虽然如此，但是这已经足够布置一个基本任务了。我们希望根据金融新闻预测股价变动。根据最终目的，有许多方法可以完成该任务。如果想根据金融新闻进行**交易**（理想情况下）就需要根据一连串新闻，提前准确地预测某公司的股价变动。然而在现实中，股价变动的因素错综复杂，而其中一些并没有在新闻报道中体现。

因此，我们将为一个比较合适的目的挖掘新闻报道——**新闻推荐**。从这个角度看，有大量的市场新闻，其中有的很有趣，而大部分则很无聊。我们想用预测文本挖掘来推荐值得花时间研究的有趣新闻报道，此处的"有趣新闻报道"指的是"有可能导致股价重大变化的新闻"。

为使问题更易于处理（实际上，这个任务既是一个很好的问题界定范例，也是一个很好的文本挖掘范例），我们必须将其进一步简化。以下是一些问题及其简化假设。

(1) 提前很长时间预测新闻效果是很难的。由于股票太多，因而新闻发布会很频繁，而市场会随之快速做出反应。举个例子，根据今天发布的新闻，预测股价一周之后的变动就并不现实。因此，我们将尝试预测新闻报道**当天**对股价的影响。

(2) 准确预测股价是很难的，因此我们只需正确预测股价变化**方向**：上涨、下跌或不变。事实上，我们将进一步把方向简化成**变化**和**不变**。这样的简化对示例应用非常有效，如果某新闻报道有可能引发或预示股价变动，那么我们就推荐它。

(3) 预测股价的微小变动是很难的，因此我们将只预测**较大**的变动。这样虽然会让所获得的事件数减少，但会使得信号更加清晰。因此，我们将故意忽略微小的变动。

(4) 将股价变动与特定的新闻报道关联起来是很难的，原则上，任何新闻都有可能影响股价。如果接受这个理念，就会遇到一个很严重的信任分配问题：怎样从今天上千条新闻中确定哪一条是相关的呢？因此我们必须缩小"因果半径"。

我们将假设，股价仅受那些提及这支股票的新闻影响。当然，这并不正确，因为虽然企业会受竞争者、顾客和客户的行为影响，但是很少有新闻能提及全部因素。但是在初次尝试中，这种简化假设可以接受。

还有一些细节需要确定。思考上面第 3 条："较大"的变化是什么？我们可以（有些随意地）把阈值定为 5%，如果股价涨幅不低于 5%，就称为**飙升**；如果跌幅不低于 5%，就称为**暴跌**；若介于两者之间，就称为**稳定**。但这样有点过于严格了，因为 4.9% 的变化和 5% 的变化区别并不大。因此，我们将指定一些"灰色区域"，使类之间更加可分（见图 10-6）。股价只有变化幅度处于 2.5% 到 −2.5% 之间，才称为**稳定**；如果变化幅度处于 2.5% 到 5% 或 −2.5% 到 −5% 之间，则不予标记。

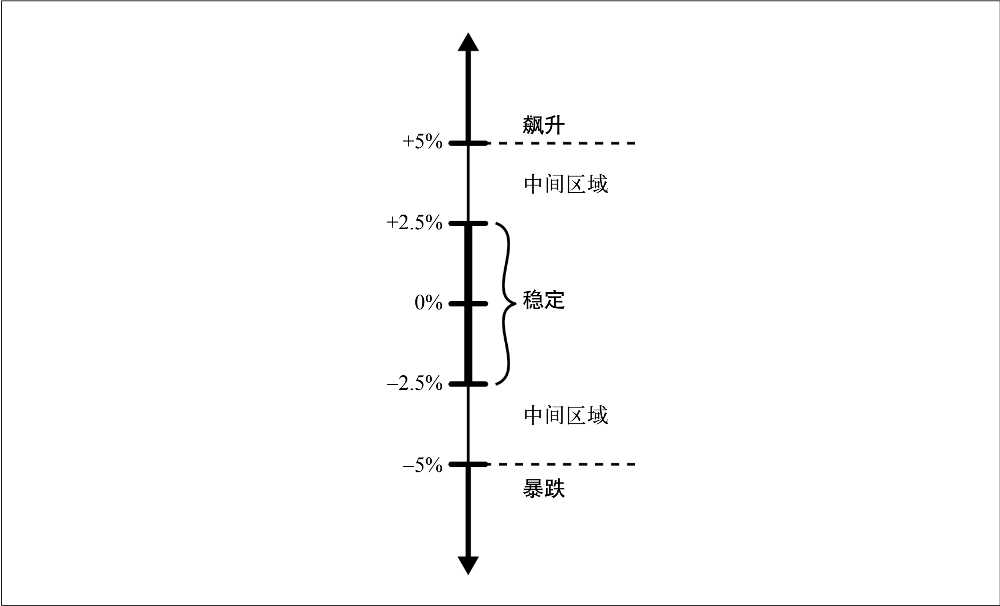


图 10-6：股价变化百分比与对应的标签

根据本例的目的，我们将构造一个二元分类问题，将"飙升"和"暴跌"合并为**变化**，作为正类，将**稳定**（**不变**）作为负类。

10.7.2 数据

我们即将使用的数据中包含两个单独的时间序列：新闻报道（文本文档）的时间序列和其对应的每日股价的时间序列。互联网上有很多金融数据源，如谷歌金融和雅虎金融。举个例子，如果你想找苹果计算机公司的相关新闻报道，那么查看雅虎新闻的网页（http://finance.yahoo.com/q?s=AAPL）即可。雅虎集合的新闻报道来自众多新闻源，如路透社、网络公关和福布斯。历史股价信息则可以通过很多信息源获取，如谷歌金融。

我们要挖掘的数据是 1999 年起纽约证券交易所和纳斯达克列出的股票的历史数据，该数据之前曾用于另一个研究（Fawcett & Provost, 1999）。该数据包含股票在主要交易所的开盘价和收盘价，以及一整年的财经新闻的大型纲要，共有近 36 000 篇。以下是语料库中的一篇新闻样本：

> 1999-03-30 14:45:00 WALTHAM, Mass. ——（美国商业新闻社）——1999 年 3 月 30 日——Summit 科技有限公司（纳斯达克代码：BEAM）[3] 与 Autonomous 技术公司（纳斯达克代码：ATCI）[4] 近日宣布，Summit 科技收购 Autonomous 技术的联合代理 / 招股说明书被证券交易委员会宣布有效。文件复印件已送至两家企业股东手中。“我们很高兴看到这些代理材料生效，也期待 4 月 29 日的股东大会。”Summit 科技的首席执行官 Robert Palmisano 说。

与许多原始文本一样，该文本中的材料非常混杂，因为它是为人类阅读而写，而非为机器解析而写（更多细节可见后文中“杂乱的新闻”）。其中包含日期和时间、新闻来源（路透社）、股票代码和链接（纳斯达克代码：BEAM），以及许多与新闻关系不大的背景材料。我们为这样的新闻标注其中提及的股票的标签。

杂乱的新闻

金融新闻语料库实际上比这篇报道杂乱得多，原因有如下几个。

首先，金融新闻种类广泛，包括收益报告、分析师评估（“我们要重申对苹果的‘强力买入’评级”）、市场评论（“今晨的其他市场推手股票包括 Lycos 公司和 Staples 公司”）、证券交易委员会档案、财务资产负债表等。企业出现在报道中的原因有很多，而一篇文档（新闻）可能会包含当日许多无关新闻的导语。

其次，新闻格式多种多样，有的新闻是列表数据，有的则是多段“今日头条新闻”的格式，不一而足。文中的含义要根据上下文理解，而文本处理系统可做不到这一点。

最后，股票标签并不完美。可能是由于标签标注过于自由，导致某些新闻即使没有提到某股票，与该股票相关的新闻推送也会包括该新闻。一个极端例子是，美国博客主 Perez Hilton 用“cray cray”来表达“疯狂”或“恶心”的含义，竟然导致他的一些博文和 Cray 计算机公司挂钩。

简而言之，如果不仔细阅读文档，那么它与股票的关联可能就不够清晰。虽然进行深入解析（或至少新闻划分）之后，文档中的一些噪声可以被消除，但词袋模型（甚至命名实体提取）并不能删除所有的噪声。

图 10-7 展示了我们希望处理的数据，它们基本上是两组相关联的时间序列。该图是 Summit 科技有限公司——一家激光视力矫正准分子激光系统制造商的股价变动图。图中一些点标注了当日新闻的标号。图的下方则是每篇新闻的总结。

注 3：下称“Summit 科技”。——译者注

注 4：下称“Autonomous 技术”。——译者注

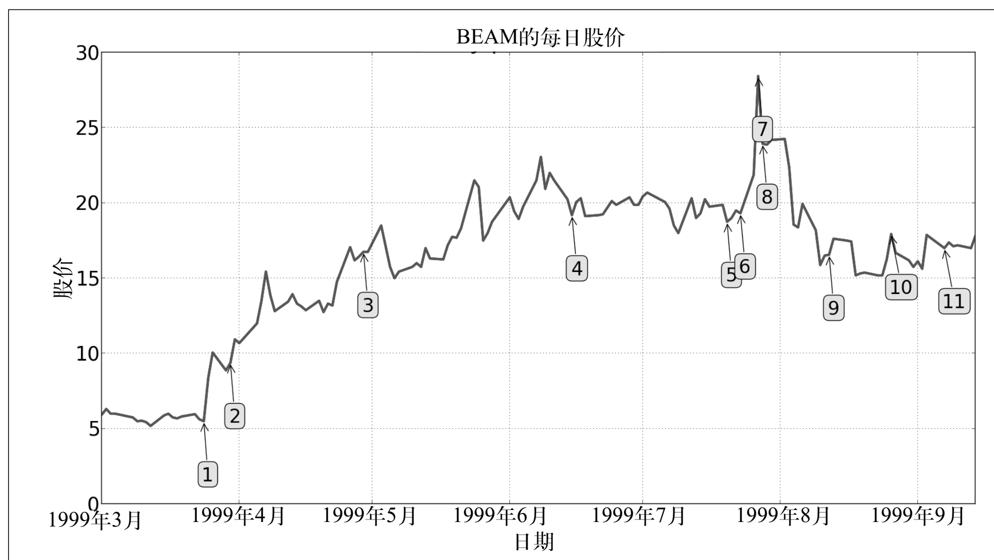


图 10-7：Summit 科技有限公司（纳斯达克代码：BEAM）的股价变动图，附加新闻摘要的注释

(1) Summit 科技宣布，截至 1998 年 12 月 31 日，三个月实现收益 2240 万美元，同比增长 13%。

(2) Summit 科技与 Autonomous 技术公司宣布，Summit 科技收购 Autonomous 技术的联合代理 / 招股说明书被证券交易委员会宣布生效。

(3) Summit 科技称其产品使用量达到第一季度新高，并确定其对 Autonomous 技术公司的收购。

(4) 宣布召开年度股东大会。

(5) Summit 科技宣布，已向证券交易委员会提交 400 万份普通股的注册声明。

(6) 美国食品及药物管理局委员会为 Summit 科技在矫正近视（无论是否散光）的 LASIK 手术中使用的激光背书。

(7) Summit 股票上涨了 $1\frac{1}{8}$，达到 $27\frac{3}{8}$。

(8) Summit 科技今日称，截至 1999 年 6 月 30 日，近 3 个月的收益涨幅为 14%……

(9) Summit 科技宣布，以每股 16 美元的价格，公开发售 350 万份普通股。

(10) Summit 科技宣布与 Sterling Vision 公司达成协议，Sterling Vision 将购买多达 6 套 Summit 科技前沿科技产品——Apex Plus 激光系统。

(11) Preferred Capital Markets 公司给予 Summit 科技有限公司“强力买入”评级，12~16 个月目标价格为 22.50 美元。

10.7.3 数据处理

如上所述，我们有两个数据流。每支股票都有当天的开盘价和收盘价，分别记录于美国东部标准时间当天上午 9:30 和下午 4:00，根据这些值，可以轻而易举地计算出价格变化的百分比。此处有一个小小的难点。我们想预测能使股价发生巨变的新闻。交易时间之外会有许多事件发生，因而靠近开盘时间和收盘时间时股价波动很大，因此我们不记录开市钟敲响时（美国东部标准时间上午 9:30）的开盘价，而记录上午 10:00 的股价，然后计算其与

下午 4:00 的股价之差。将该差值除以收盘价后，就得到了当日股价变化的百分比。

新闻报道更需要谨慎对待。我们事先给新闻报道加上了股票的标签，这些标签大部分是准确的（前文补充栏“杂乱的新闻”详细探讨了这种文本挖掘的困难之处）。几乎所有新闻都有时间标记（没有的已被删掉），因此我们可以将它们按正确的日期和交易窗口排序。因为想知道新闻与其可能影响的股票之间的紧密联系，所以我们忽略所有提及两只以上股票的新闻，这样就删除了许多单纯的总结和新闻聚合。

经过 10.3.1 节所描述的基本步骤，我们把新闻简化成了 TFIDF 表示形式。特别地，每个单词都经过大小写标准化、词干提取，并且我们删除了停用词。最后，我们构造了一个 2-grams 模型，使得新闻中每个单独的词语和相邻的两个单词都能用来表示新闻。

准备工作完成后，我们给每篇新闻打上相关的股票价格变化的标签（**变化**或**不变**），如图 10-7 所示。这样一来，就得到了约 16 000 篇可用的有标签新闻。仅供参考，在所有新闻中，75% 标为不变，13% 标为飙升，12% 标为暴跌。飙升和暴跌的新闻合并，构成**变化**分组，因此 25% 的新闻会导致相关股票价格发生巨变，而 75% 则不会。

10.7.4 结果

在深入研究该结果前，先说一点题外话。

前面的章节（尤其是第 7 章）强调，为了设计评估框架，仔细考虑要解决的商业问题极其重要。但是本例并没有经过如此仔细的考虑。如果任务的目的是触发股票交易，那么我们可以提出一个包括阈值、时间限制和交易成本的总体交易策略，并据此进行完整的成本收益分析。[5] 但现在我们的目的是推荐新闻（即回答“哪些新闻会导致股票价格发生巨变”），因为该问题非常开放，所以我们不会确切计算用于决策的成本收益。因此，期望值计算和收益图像并不适用于此问题。

我们还是来看看模型的预测能力，看看这个问题能被解决到什么程度。图 10-8 展示了三个样本分类器（逻辑回归、朴素贝叶斯和分类树）的 ROC 图像，以及一条随机分类线。这些曲线根据 10 重交叉验证的结果取平均，其正类为**变化**，负类为**不变**。许多问题显而易见。首先，因为曲线有一处明显的远离对角线（随机分类线）的“弯曲”，且 ROC 曲线下面积（AUC）全都远大于 0.5，所以新闻报道的确可以预测股价变动；其次，逻辑回归和朴素贝叶斯表现相似，而分类树模型显然差一些；最后，这些曲线没有明显的优势区间（或劣势区间）。曲线的凹凸部分有时能暴露出问题的特征，或数据表达的缺陷，但这里看不出来。

图 10-9 展示了这三种分类器对应的提升度曲线，仍根据 10 重交叉验证的结果取平均。前文提到，语料库中 1/4（25%）的新闻属于正类（即能给股价带来巨变）。每条曲线表示使用该模型对新闻进行评分和排序时，我们所能得到的精度的提升度 [6]。比如，在 $x = 0.2$ 的

注 5：一些研究者已经做过了这样的分析，通过模拟股票交易和计算投资回报来评估他们的系统。读者可参考例如 Schumaker & Chen（2010）在 AZFinText 中的文章。

注 6：回忆第 7 章，精度指的是超过分类阈值的数据项确实为正的比例，而提升度指的是上述情况比在整个总体中随机寻找的精度高多少倍。

点上，逻辑回归和朴素贝叶斯的提升度均约为 2.0，这意味着，如果对所有新闻报道打分，然后选择前 20%（$x = 0.2$）的新闻，那么在其中寻找正向新闻的精度会是在全部新闻中寻找的 **2 倍**（提升度为 2）。因而，在根据模型排序的前 20% 的新闻中，有**一半**能给股价带来巨变。

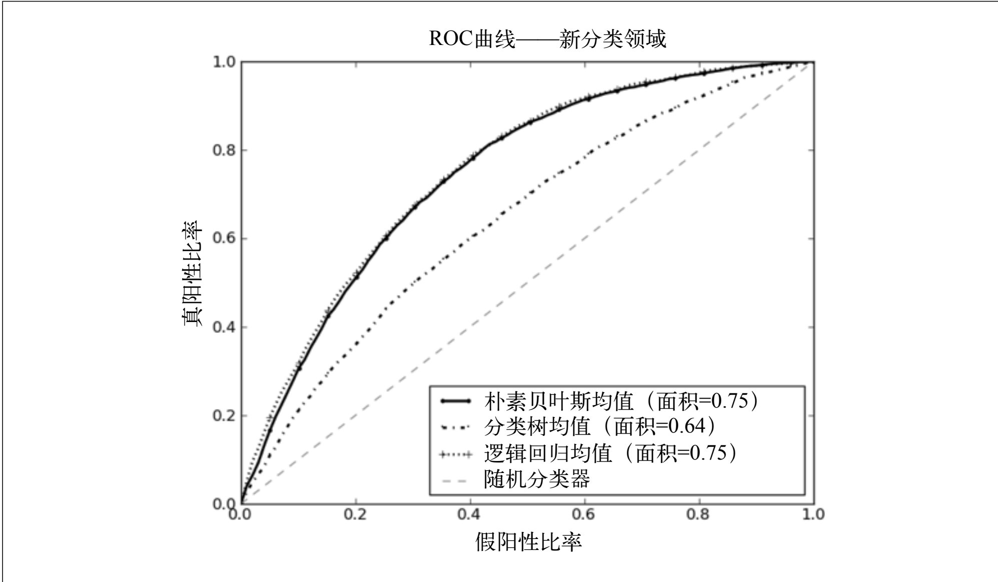


图 10-8：股票新闻分类问题的 ROC 曲线

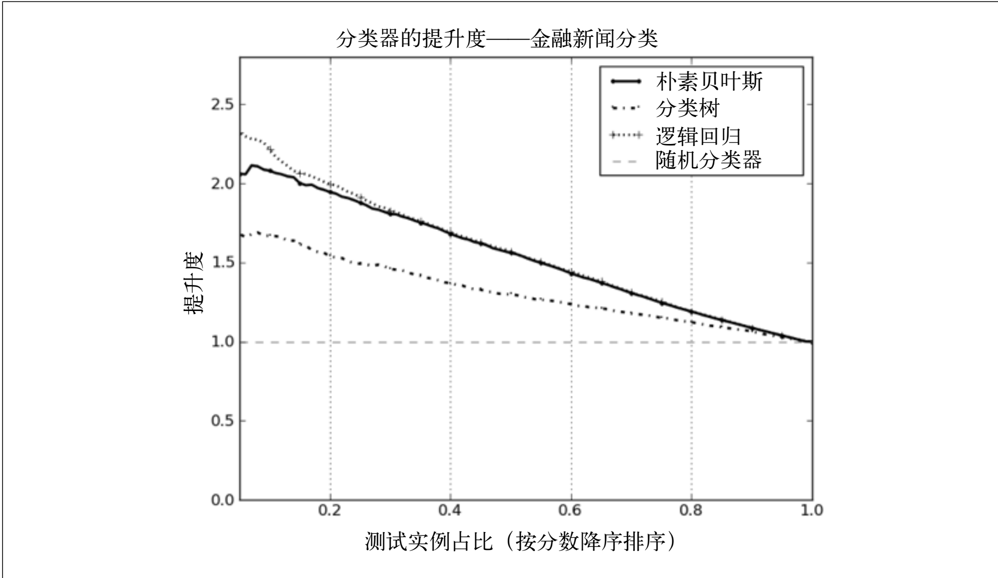


图 10-9：股票新闻预测问题的提升度曲线

在下结论前，再看看这个问题中的一些重要词语。虽然本例的目的并不是根据数据构建易理解的规则，但是 Macskassy 等人在 2001 年就在同一个语料库上做了这样的工作。以下是从他们的结论中找出的信息增益[7]高的词语，每条词语可能是单词，也可能是删除后缀（体现在括号中）后的词干：

```
alert(s,ed), architecture, auction(s,ed,ing,eers), average(s,d), award(s,ed),
bond(s), brokerage, climb(ed,s,ing), close(d,s), comment(ator,ed,ing,s),
commerce(s), corporate, crack(s,ed,ing), cumulative, deal(s), dealing(s),
deflect(ed,ing), delays, depart(s,ed), department(s), design(ers,ing), economy,
econtent, edesign, eoperate, esource, event(s), exchange(s), extens(ion,ive),
facilit(y,ies), gain(ed,s,ing), higher, hit(s), imbalance(s), index,
issue(s,d), late(ly), law(s,ful), lead(s,ing), legal(ity,ly), lose, majority,
merg(ing,ed,es), move(s,d), online, outperform(s,ance,ed), partner(s), payments,
percent, pharmaceutical(s), price(d), primary, recover(ed,s), redirect(ed,ion),
stakeholder(s), stock(s), violat(ing,ion,ors)
```

其中有些词暗示关于企业或其股价的重大好消息或坏消息，有些词（econtent、edesign、eoperate）同时也暗示 20 世纪 90 年代末的“互联网热潮”，而这个语料库正是生成于那个 e 前缀流行的时代。

虽然本例是本书中最复杂的示例之一，但其中挖掘金融新闻的方法仍颇为简单。本例可以通过许多方法来扩展完善。词袋表示法就是本例的第一选择，命名实体识别也能用于更好地提取新闻中包含的企业名和人名。更好的是，事件解析能发挥重要作用，因为新闻报道通常报道的是事件，而不是企业的静态事实。因为单个单词不能明确地体现事件的主体和客体，而一些重要的修饰语，如 not、despite 和 expect，也不一定与它们修饰的短语邻近，所以词袋模型在此问题中处于劣势。最后，在计算股价变动时，我们仅考虑了当天的开盘价和收盘价，而不是每小时或即时（秒级）的股价变化。市场对新闻的响应极快，如果想根据信息进行交易，就需要股票价格和新闻报道都含有详细可靠的时间标记。

关于根据金融新闻预测股价的早期作品

过去 15 年，有不少人研究过将金融新闻报道与市场活动相关联的问题，甚至本书作者也有一些相关的早期作品（Fawcett & Provost, 1999）。因为大部分早期作品在数据挖掘之外的领域发表，所以数据挖掘社区很可能还不知道这个问题和相关作品。我们将在此提及几篇文章，以便感兴趣的读者进一步研究。

Mittermayer 和 Knolmayer 的调查（2006）是一个合适的开始，虽然现在看来有些过时，但它仍提供了到当时为止还算不错的方法综述。

大多数研究者会认为该问题是根据新闻预测股票市场，而本章中我们反其道而行之，根据新闻的未来影响来推荐新闻。这样的任务被 Macskassy 等人（2001）称为**信息分类**。

早期的作品关注主流媒体中金融新闻的影响，而后期的作品则会把互联网中其他来源的评论观点也考虑进去，比如 Twitter 更新、博客帖子和搜索引擎趋势等。Mao 等人（2011）发表的论文就对这些额外信息源的影响进行了仔细的分析和比较。

注 7：可回忆第 3 章内容。

最后，虽然该作品中的问题本身并非文本挖掘，但我们还是要提一下 Cohen、Diether 和 Malloy 的 *Legislating Stock Prices*（2012）。他们研究了政客、法律和受法律影响的公司之间的关系。显然，三者存在关联且互相影响，但令人感到意外的是，这样的关联竟然没有被华尔街发现。基于公开数据，研究人员发现了一种“对公司股价简单的、之前未被察觉的影响”，并将其发表了出来，以供交易盈利。这让我们不由想到，还有许多潜藏的关系等待我们去发现。

10.8 小结

在实际问题中，我们获得的数据有时不是用简洁的特征向量表示，无法直接作为大多数数据挖掘算法的输入。因此，实际问题通常需要经过一些数据表示工程的处理，才能够实施数据挖掘。一般来说，比较简单的方法是把数据转化成符合现有工具要求的形式。文本、图像、音频、视频和空间信息形式的数据通常需要特别处理，有时甚至需要数据科学团队具有一定的专业知识。

本章探讨了一种需要预处理的常见数据形式：文本。将文本转化为特征向量的一种常用方法是，把每篇文档分解为单词（即“词袋”表示法），然后用 TFIDF 公式给每个词语赋值。该方法相对简单、成本低廉且用途广泛，而且几乎不需要领域知识（至少一开始不需要）。该方法虽然简单，但在多种问题中的表现却惊人地好。在第 14 章，我们将在一个完全不同的非文本问题中回顾这些概念。

第11章

决策分析思维（二）：面向分析工程

基本概念：用数据科学解决商业问题，首先要进行分析工程，即根据现有的数据、工具和技术，设计分析解决方案

示例方法：将期望值作为数据科学解决方案设计框架

数据科学的本质是根据原则性技术，提取数据中的信息或知识。但是正如本书一直探讨的，我们很难把技术与重要的商业问题完全匹配，也很难得到能直接应用于技术的数据。讽刺的是，商业人士通常比数据科学新手更容易接受这个事实（这对前者来说往往是显而易见的）因为在统计学、机器学习和数据挖掘等领域的教学过程中，学生们面对的问题通常都可以用他们所学的工具直接解决。

但现实问题往往要复杂得多。商业问题很少是单纯的分类问题、回归问题或聚类问题，它们就是商业问题。回顾数据挖掘流程第一个环节中的小循环。这里我们主要关注的是业务理解环节和数据理解环节。在这些环节中，我们必须**设计**或**策划**出商业问题的解决方案。与广义工程中一样，数据科学团队除了要理解商业需求，还需要理解用来解决问题的工具。

本章将用两个案例来说明这种**分析工程**。在这两个案例中，我们将看到本书中反复出现的基本原则和一些已经介绍过的具体技术。贯穿这些案例的一个共同主题是，期望值框架（回忆第 7 章）如何把商业问题分解成几个子问题，从而让我们能够用经过实践检验的数据科学技术将它们一一击破，随后，期望值框架还会帮我们把结果组合成原问题的解决方案。

11.1 为慈善机构寻找最佳捐赠人

我们有一个应用数据科学原则与技术的经典商业问题：目标市场营销。它是一个非常好的教学案例，原因有两个。第一，许多行业都存在类似目标市场营销的问题，比如传统的目标（数据库）市场营销、用户专用优惠券的发放、在线精准广告等；第二，目标市场营销问题的基本结构也出现在许多其他问题中，比如先前反复讨论的用户流失管理问题。

在本次案例研究中，请思考一个目标市场营销的真实示例：为慈善机构寻找邮件最佳发送对象。募捐组织（包括大学中的）需要权衡预算和潜在捐款人的容忍度，在每次募捐活动中，他们都会向一群"慷慨的"捐款人募捐。这种募捐活动可能所针对的捐款人数量较多，但是成本较低，不太频繁，也可能所针对的捐款人数量较少，但针对性更强，激励方案的成本更高。

11.1.1 期望值框架：分解商业问题，重组解决方案

我们通常需要为问题"策划"一个分析方案，而基本概念给我们提供了分析框架。为了建立数据分析式的思维方式，我们首先要通过数据挖掘（第 2 章）流程来建立全面的分析架构：从业务理解环节和数据理解环节开始。具体来说，关注点应该始终停留在基本原则之一：我们究竟想解决什么商业问题（见第 7 章）？

让我们来把问题具体化。数据挖掘工程师可能会立马想到：要根据每个潜在用户（本例中也就是潜在捐款人）是否会对优惠做出响应进行建模。但是仔细考虑过这个问题后，你会发现，本例中的响应情况分为很多种：有的人可能会捐 100 美元，而有的人可能只捐 1 美元。我们必须把这些情况都考虑进来。

我们是否希望捐款总额最大化？（这里的捐款总额既可以指一次捐款活动中的捐款金额，也可以指捐款人一生之中所有捐款的金额，为了简化问题，本章选择前一种情况。）如果实现目标的手段是针对许多人，每个人仅捐 1 美元，而成本正好也是每人 1 美元呢？这样我们将几乎筹集不到钱。因此，我们需要重新思考这个问题。

关注需要解决的商业问题本身有助于迅速得到答案，因为对于精通商业的人来说，这是显而易见的：我们需要将捐款**利润**，也就是考虑成本后的净利润最大化。不过，即便掌握了估计响应概率的方法（这很明显是对二元结果的类概率估计的应用），我们依然不知道如何估计收益。

这里再次强调，那些基本概念有助于组织思维并策划数据分析解决方案。在应用另一条基本概念后，就能用期望值的框架来组织数据了。我们可以把第 7 章介绍的概念应用到问题定义中，把期望值作为策划问题解决方案的框架。请回忆目标用户 $\boldsymbol{x}$ 的期望收益（或成本）公式：

$$\text{目标用户的期望收益} = p(R \mid \boldsymbol{x}) \cdot v_R + [1 - p(R \mid \boldsymbol{x})] \cdot v_{NR}$$

其中，$p(R \mid \boldsymbol{x})$ 是用户 $\boldsymbol{x}$ 的响应概率，v_R 是响应的值，而 v_{NR} 是未响应的值。因为每个人要么响应、要么不响应，所以不响应的概率估计值是 $1 - p(R \mid \boldsymbol{x})$。正如第 7 章所讨论的，本书中许多技术可以通过挖掘历史数据来模拟概率。

然而，期望值框架让我们知道，商业问题和目前我们考虑过的问题有些不同。本例中捐款人给出的值各不相同，在把某捐款人作为目标之前，谁也不知道目标捐款人的捐款额会是多少！为了明确这一点，我们来修改一下公式：

$$\text{目标捐款人的期望收益} = p(R \mid \boldsymbol{x}) \cdot v_R(\boldsymbol{x}) + [\,1 - p(R \mid \boldsymbol{x})\,] \cdot v_{NR}(\boldsymbol{x})$$

其中 $v_R(\boldsymbol{x})$ 是捐款人 $\boldsymbol{x}$ 响应时得到的值，而 $v_{NR}(\boldsymbol{x})$ 是用户 $\boldsymbol{x}$ 未响应时得到的值。响应值 $v_R(\boldsymbol{x})$ 是收到的捐款减去募捐成本，而未响应的值 $v_{NR}(\boldsymbol{x})$ 就是 0 减去募捐成本。为了考虑得更全面，我们还需要估计**不**把捐款人作为目标的收益，然后通过比较两者来决定是否将其作为目标。非目标用户的期望收益很简单，就是 0——在本例中，我们认为捐款人不会不经请求就自发地捐款。虽然事实上并非总是如此，但在这里姑且这样假设。

为什么期望值框架会对我们有帮助呢？因为有了这个框架就能通过数据估计出 $v_R(\boldsymbol{x})$ 和 / 或 $v_{NR}(\boldsymbol{x})$ 了。其实通过回归建模也能估计这些值。根据目标捐款人的历史数据，可以用回归建模来估计捐款人的响应值。不过，期望值框架还能给我们一个更确切的方向：$v_R(\boldsymbol{x})$ 是所预测的**捐款人会响应**时的值，它可以通过只有响应用户的数据集训练出的模型来预测。事实证明，这比普遍地估计目标捐款人的响应值有效多了，因为绝大多数的捐款人根本不会响应。而回归建模则需要区分未响应情况下的 0 值和捐款额较小导致的极小值。

再回顾一下前面的内容。期望值框架有助于分解商业问题的原因正如第 7 章所描述，期望值是每种情况的概率和它对应值的乘积的总和，而数据科学提供的方法正好能让我们估计概率和其对应值。需要明确的是，尽管我们可能不需要估计其中一些量（如 $v_{NR}(\boldsymbol{x})$，本例中假设它永远为 0），然而准确估计它们确实是一件很重要的事情。期望值框架可以有助于把很复杂的商业问题分解成一个个子问题，以便于寻找解决方案。最后它还能告诉我们如何将这些子问题组合到一起。在该示例（选用它是因为推导简单）中，直观答案非常令人满意：给那些期望捐款额高于信息成本的捐款人发信息！数学上，我们只需要寻找那些期望收益大于 0 的捐款人，这在代数上简化不等式即可。设 $d_R(\boldsymbol{x})$ 为用户 $\boldsymbol{x}$ 响应时的期望捐款额，c 为发信息的成本，那么：

$$\text{目标捐款人的期望收益} = p(R \mid \boldsymbol{x}) \cdot v_R(\boldsymbol{x}) + [\,1 - p(R \mid \boldsymbol{x})\,] \cdot v_{NR}(\boldsymbol{x})$$

我们希望收益大于 0，从而：

$$p(R \mid \boldsymbol{x}) \cdot (d_R(\boldsymbol{x}) - c) + [1 - p(R \mid \boldsymbol{x})] \cdot (-c) > 0$$
$$p(R \mid \boldsymbol{x}) \cdot d_R(\boldsymbol{x}) - p(R \mid \boldsymbol{x}) \cdot c - c + p(R \mid \boldsymbol{x}) \cdot c > 0$$
$$p(R \mid \boldsymbol{x}) \cdot d_R(\boldsymbol{x}) > c$$

也就是说，期望捐款额（左侧）应大于募捐成本（右侧）。

11.1.2　简短的题外话：选择性偏差

这个例子引出了数据科学的另一个重要问题，虽然其处理方法已经超过了本书的范围，但此处还是有必要作简短的讨论。请注意，在捐款预测的建模过程中，数据可能是有偏的——也就是说，样本并不是从所有捐款人中随机抽取的。为什么呢？因为这些数据来自于以前的捐款活动，即来自于以前**的确响应过**的捐款人。这与根据信贷用户的历史数据模拟资信水平的思路很相似：他们是你过去认为信用良好的那些用户！但是，你想用模型找

到的是整个总体中将来最有希望捐款的人，那么过去碰巧被选中的那些人为什么就是整个总体中的好样本呢？这就是一种**选择性偏差**，数据不是从需要实际应用模型的总体中随机选择的，相反它在某种程度上是有偏的（比如有人只是偶然去捐款，比如根据过去的方法选择目标用户，再比如有人只是过去被授信）。

数据科学家面临的一个重要问题是：使数据产生偏差的选择过程是否会影响目标变量的值？在信用风险建模中，答案毫无疑问是**肯定的**，因为老顾客正是因为信用良好被挑选了出来。虽然捐款的例子可能不会那么直观，但是我们推测，捐款额较大的捐款人捐款频率通常并不高。比如，有人可能在每次收到捐款请求时捐出 10 美元，而另一些人则会一次性捐出 100 美元，然后不管之后再看到多少次这样的捐款活动，都觉得自己一段时间内不用再捐了。于是结果就会产生偏差：某些过去恰巧参加了一些捐款活动的人可能更偏向于捐得少的人。

不过幸运的是，有的数据科学技术能够帮助建模者处理这样的选择性偏差。但这些技术同样也超出了本书范围，感兴趣的读者不妨从 Zadrozny 的作品（Zadrozny & Elkan, 2001; Zadrozny, 2004）读起，了解在这个募捐案例中处理选择性偏差的方法。

11.2　更复杂的用户流失示例回顾

接下来让我们在用户流失示例中应用所学的知识，从数据分析的角度进行研究。在先前的尝试中，我们并没有竭尽全力全面地处理这个问题。当然，这是故意这样设计的，因为当时我们还没有学完所有要用的知识，而且那些不全面的尝试足以说明问题。但是现在我们可以利用刚才募捐问题涉及的基本数据科学的概念，更加详细地研究这个问题。

11.2.1　期望值框架：构建更复杂的商业问题

首先，我们要解决的商业问题究竟是什么？保持示例问题的基本设定：我们的电信公司用户流失严重，市场部为此策划了特别的优惠来留住用户，而我们的任务就是把优惠活动有针对性地提供给合适的用户。

最初，我们决定要用数据找到最有可能在合约到期后（短时间内）离开公司的用户。进一步地，我们要关注一下哪些用户的合约即将到期，因为大多数流失就发生于这段时间。不过，我们真的想要把优惠提供给那些最有可能离开公司的用户吗？

这就需要回到基本概念上来：我们要解决的商业问题究竟是什么？为什么用户流失是个问题？因为公司会因此赔钱，所以真正的商业问题是赔钱。如果公司在一个用户身上赔得比挣得多，那么就算他流失了也无所谓。我们想要做的是限制损失的金额，而不是简单地留住大部分用户。因此，像捐款问题一样，我们要把用户的**价值**也考虑进来。此时期望值框架就能帮助设计这样的分析，其过程与上面类似。在流失案例中，每个人的值更容易估计。因为这是本公司自己的用户，而公司有他们的账单记录，所以通过对以前的值应用外推法，就可以非常准确地预测他们的未来收益值（取决于这些用户是否留存）。但是在这个案例中，我们还没能完全解决问题，通过对期望值的设计分析我们会知道原因。

我们将用期望值框架来深入探究数据挖掘流程中业务理解和数据理解这两个环节。我们是否可以把这个案例和捐款案例做同样的处理？像捐款案例中一样，我们可以把给目标用户

特殊优惠的期望收益表示成如下形式：

$$目标用户的期望收益 = p(S \mid \boldsymbol{x}) \cdot v_S(\boldsymbol{x}) + [1 - p(S \mid \boldsymbol{x})] \cdot v_{NS}(\boldsymbol{x})$$

其中，$p(S \mid \boldsymbol{x})$ 是用户 $\boldsymbol{x}$ 在作为目标用户留在 [1] 公司的概率，$v_S(\boldsymbol{x})$ 是用户 $\boldsymbol{x}$ 留在公司时我们得到的值，而 $v_{NS}(\boldsymbol{x})$ 则是用户 $\boldsymbol{x}$ 不留在公司（离开或流失）时我们得到的值。

我们是否能用这个公式来选择提供特殊优惠的目标用户呢？其他条件不变的情况下，选择值最高的用户似乎就是选择最有可能**留下**的用户，而不是最有可能离开的用户！为了讲得更明白，让我们来简化一下这个例子，假设用户流失的值为 0，那么期望值公式就变成了：

$$目标用户的期望收益 = p(S \mid \boldsymbol{x}) \cdot v_S(\boldsymbol{x})$$

这与我们想把最有可能离开的用户作为目标的初衷不一致。可是问题究竟出在了哪里呢？期望值框架告诉我们：我们还需要更谨慎。我们不想草草应用之前在捐款问题中的做法，而想仔细考虑一下现在这个问题。我们并不想把将会留存的高价值用户作为目标，而是想把流失后造成的损失最多的用户作为目标。这是个复杂的问题，而期望值框架不仅有助于系统思考，还会启发我们解决问题。在捐赠问题中，我们曾说："为了考虑得更全面，我们还需要估计不把捐款人作为目标的收益，然后通过比较两者来决定是否将其作为目标。"当时我们之所以允许自己忽略这一点，是因为我们假设捐款人不会在没有受到募捐请求的情况下自发捐款。但是，在业务理解环节，我们必须考虑到商业问题的每个细节。

考虑一下用户流失问题中"不作为目标"的情况：如果不将用户作为目标，其值是否为 0？这可不一定。如果用户不被选为目标却还是会留下，那么其实我们能取得更高的值，因为我们没有在激励上花费成本！

11.2.2 评估激励的影响

让我们继续深入研究。首先，分别计算将用户选为优惠激励目标的收益和不将用户选为目标的收益，并详细定义一下激励成本。假设 $u_S(\boldsymbol{x})$ 是用户 $\boldsymbol{x}$ 留下的收益，不包含激励成本；$u_{NS}(\boldsymbol{x})$ 是用户 $\boldsymbol{x}$ 离开的收益，同样不包含激励成本。为进一步简化问题，假设无论用户留下还是离开，我们都需要承担激励成本 c。



对流失问题来说这并不完全真实，因为激励通常根据用户是否流失包含了不同的成本构成，比如一部新手机。对这个小问题展开分析也是非常简单的，我们也可以得到相同的定性结论。你不妨一试。

那么，我们来分别计算把用户作为目标和不把其作为目标的期望收益值。在此需要说明，用户留下和离开的概率估计根据其是否被选为目标（但愿会）而存在差异（希望激励会起作用）。我们在两种情况下（选为目标，T，或不选为目标，$notT$）分别表示留下的概率。选为目标的期望收益是：

$$EB_T(\boldsymbol{x}) = p(S \mid \boldsymbol{x}, T) \cdot (u_S(\boldsymbol{x}) - c) + [1 - p(S \mid \boldsymbol{x}, T)] \cdot (u_{NS}(\boldsymbol{x}) - c)$$

不选为目标的期望收益值是：

注 1：Stay，取首字母。——译者注

$$EB_{notT}(\boldsymbol{x}) = p(S \mid \boldsymbol{x}, notT) \cdot u_S(\boldsymbol{x}) + [1 - p(S \mid \boldsymbol{x}, notT)] \cdot u_{NS}(\boldsymbol{x})$$

现在，为了完善商业问题的定义，我们要把那些选为目标后会带来最大收益的用户，也就是那些 $EB_T(\boldsymbol{x}) - EB_{notT}(\boldsymbol{x})$ 最大的用户选为目标。这其实是一个比以前更复杂的问题——但期望值框架能帮我们组织思路，有助于我们系统地思考并精确的针对目标进行分析。

同时，期望值框架还能让我们看到现在这个问题与之前研究过的问题在结构上的区别。尤其是，我们要考虑**不把用户选为目标**的后果（分析 EB_T 和 EB_{notT}），以及激励的实际影响（即 EB_T 和 EB_{notT} 的差值）。[2]

我们再简短地用一些有关数学的题外话来说明这个问题。试想“选作目标的值” $VT = EB_T(\boldsymbol{x}) - EB_{notT}(\boldsymbol{x})$ 达到最大的情况。在如果用户不留下，公司就将一无所获的假设下，我们来展开并简化 VT 的公式。

公式 11-1：*VT* 分解

$$\begin{aligned} VT &= p(S \mid \boldsymbol{x}, T) \cdot u_S(\boldsymbol{x}) - p(S \mid \boldsymbol{x}, notT) \cdot u_S(\boldsymbol{x}) - c \\ &= [p(S \mid \boldsymbol{x}, T) - p(S \mid \boldsymbol{x}, notT)] \cdot u_S(\boldsymbol{x}) - c \\ &= \Delta(p) \cdot u_S(\boldsymbol{x}) - c \end{aligned}$$

其中 $\Delta(p)$ 是将用户选为目标和不将其选为目标时，用户留在公司的概率预测值之差。我们再一次看到了直观的结果：根据反映用户是否将要留存的期望值，我们希望将那些留存概率变化最大的用户作为目标用户！换句话说，就是把选为目标后期望值变化最大的用户作为目标。（$-c$ 对本示例背景下的每个用户而言都相同，公式包含这个量仅仅是为了确保 VT 不会成为经济损失。）

千万不要忘记，这些工作都属于业务理解环节。接下来看看它对数据挖掘流程中其他部分的影响。

11.2.3 从期望值分解到数据科学解决方案

前面的讨论，尤其是公式 11-1 强调的分解，在数据理解、数据定义、建模和评估等方面为我们提供了指导。特别是通过分解，我们可以明确要构建的模型，即用来估计 $p(S \mid \boldsymbol{x}, T)$ 和 $p(S \mid \boldsymbol{x}, notT)$ 的模型。两者分别为用户在被选为目标的情况下和不被选为目标的情况下留在公司的概率。与先前的数据挖掘解决方案不同，在这里我们要构造两个独立的概率估计模型。一旦这些模型建立，我们便可以用它们计算目标用户的期望值。

重要的是，期望值的分解能使我们在数据理解环节的努力更集中。我们需要什么数据来构建这些模型？在两种情况下，我们都需要合约已到期的用户样本。实际上，我们需要的是合约到期已经很长时间，是走还是留已经非常确定的用户样本。第一个模型需要的是被选为特殊优惠目标的用户样本，而第二个模型则需要**未**被选为目标的用户样本。这些样本可能可以代表模型要应用的用户群（详见上文对选择性偏差的讨论）。为了深入理解数据理

注 2：这也是因果分析的一个基本出发点：构建一个所谓的反事实情景，以评估两种相同场景下期望值的差异。类比医疗诊断中需要评估治疗的因果影响时的情况，这些场景通常被称作“治疗”情况和“未治疗”情况。因果分析的不同框架，从随机试验到回归因果分析，再到更现代的因果建模方法，本质上都存在这种期望值的差值。第 12 章将更深入地讨论因果数据分析。

解环节，请更深入地思考一下这两者。

如何获得未被选为特殊优惠目标的用户的样本呢？首先，我们需要确定业务环境没有发生本质变化，否则会影响用历史数据预测用户流失的有效性（比如 iPhone 对 AT&T 用户的独家发售，就会对其他通信公司造成这样的情况）。假设不存在这样的变化，那么收集所需数据就会相对简单：电信公司会保留大量用户数据长达几个月，用于开具账单、欺诈检测和其他一些目的。既然这是一种新的优惠，那么也就没有用户曾被选为该优惠活动的目标。我们还需要仔细检查，以确认这些用户没有接受其他优惠活动，从而保证用户流失概率不受影响。

模拟 $p(S|\boldsymbol{x}, T)$ 的情况则大相径庭，而且再一次强调了期望值框架有助于尽早理清思路，突出当前面对的问题和挑战。其中的困难是什么呢？困难是，因为优惠是全新的，没人享受过，所以我们找不到相关数据来构建模型，也就无法估计 $p(S|\boldsymbol{x}, T)$！

然而，因为一些业务上的紧急事件，我们急需减少用户流失，所以我们还是要硬着头皮往下进行。市场部对这次特殊优惠胸有成竹，而我们当然也有一些数据可以告诉我们该如何往下进行。这种情景在解决实际业务问题的数据挖掘应用中并不少见。期望值的分解可以帮助我们得到一个复杂的公式，它有助于更好地理解问题。但是我们可能不愿意或没有能力处理这样复杂的公式，这可能是因为手头没有资源（数据、人力或计算能力）。在用户流失示例中，所缺少的是必要的数据。

还有一种情景是，我们不相信公式中后来添加的复杂项能大幅提升效率。比如，我们可能会做出这样的推断："是，公式 11-1 让我知道了该怎么做，但我觉得用更简单、成本更低的公式也能做到这点。"举个例子：如果假设用户一旦被提供优惠就一定会留在公司（即 $p(S|\boldsymbol{x}, T) = 1$），会怎么样呢？虽然这个假设显然过度简化了问题，但是并不影响我们采取措施——而且在实际业务中，我们必须准备好在没有理想信息的情况下采取措施。你可以用公式 11-1 证明，应用这种假设的结果不过是把 $1 - p(S|\boldsymbol{x}, notT) \cdot u_S(\boldsymbol{x})$ 取最大值的用户（也就是如果其离开，公司将蒙受最大期望损失的用户）选为目标。如果我们没有关于优惠活动所产生的不同实际影响的数据，那么这样做也很有道理。

在建模目标数据不足的情况下，还有另外一种做法：用目标标签的"替代品"来标记数据。比如，市场部可能曾经推出过一种相似但不完全相同的优惠，如果向用户推出优惠的情景也相似（回忆上文讨论的选择性偏差问题），那么不妨用替代标签来建模。[3]

期望值分解还强调了另一种选择。为 $p(S|\boldsymbol{x}, T)$ 建模需要什么？需要获取数据，特别是获取目标用户的数据。因此我们需要把用户选为目标，但这会带来一定成本。如果因为目标选得很糟糕，结果在那些响应概率低的用户身上浪费了成本，怎么办？这种情况关系到数据科学的第一条基本原则：数据应该被作为一项资产来处理。我们不仅要考虑如何利用已有资产，还要考虑如何投资资产，以得到高额回报。回忆 1.7 节中 Signet 银行面临的情况。因为他们没有关于用户对他们所设计的多种新优惠的不同响应的数据，所以他们对数据进行了投资，尽管由于广泛推广优惠活动而承受了一些损失，然而他们所取得的数据资产却正是他们成为成绩辉煌的 Capital One 的原因。我们面临的情况可能不涉及那么大范围，因

注 3：在一些应用中，替代标签可能来自与实际目标标签所在事件完全不同的事件。比如，在预测看到有针对性的广告之后，客户是否会购买产品时，实际上有关购买量变化的数据非常稀少，而把广告品牌网站访问量作为购买量的替代变量进行建模就惊人地有效。

为我们只有一种优惠。而在提供优惠时损失的钱数，也不太可能像在用户欠款时，Signet 银行损失得那么多。但无论如何，所学到的东西是一样的：如果愿意在关于人们如何响应优惠的数据上花成本，我们就能更好地针对未来的用户提供优惠。



我们有必要重申一下深入了解业务的重要性。根据优惠的结构来看，即使用户不接受优惠，我们可能也不会损失多少，因此更简单的公式可能就足够了。

注意，不只对数据资产的投资需要谨慎进行，应用本书中提及的概念工具也是如此。回忆第 8 章中用学习曲线来将模型性能可视化的概念，学习曲线有助于理解数据量，即本例中到目前为止对数据的投资额，和相应的泛化能力提升的关系。我们可以轻松地扩展泛化能力的概念，以囊括相对于基线的性能提升（回忆基本概念：仔细考虑你将拿什么作比较），其中基线可以是备选的简单用户流失模型。因此，我们会慢慢地对数据进行投资，以观察更大的数据量是否会带来更好的性能，以及曲线外凸是否代表了更大的提升空间。如果通过分析发现投资并不划算，那么我们可以停止投资。

重要的是，这并不意味着这些投资就白白浪费了。我们投资的目标是信息，此处即额外的数据是否能帮助我们有效并且合算地减少用户流失的信息。

此外，用期望值定义问题还能扩展问题的定义，从而提供一种结构化的方法来解决以下问题：**最佳优惠额是多少**？我们可以拓展定义，使之包含多种优惠，并判断哪种优惠能使用户的收益达到最大。或者，我们也可以把优惠设为参数（比如用一个可变的折扣额），然后以得到最佳期望值的折扣额为目的进行优化。当然，这样可能会带来额外的数据收集成本，因为这需要通过实验来判断不同用户在不同优惠水平下去留的概率。这同样与 Signet 银行变成 Capital One 的过程中所做的努力类似。

11.3　小结

通过关注捐款示例和用户流失示例，我们知道了期望值框架如何能够帮助辨明真正的商业问题，也了解了数据挖掘在该问题的解决方案中所扮演的角色。

我们可以继续详细讨论更多商业问题的细节，并发现问题潜在的复杂性（和对解决方案的更高要求）。你可能会问："这什么时候是个头？我总不能一直分析下去吧？"原则上，你得一直分析下去。但建模通常需要进行一些简化假设，以保证问题易于处理。在分析工程中通常有下面几点可供参考：

- 我们无法从这个事件中获取数据；
- 在这方面准确建模的成本太高；
- 该事件太不可能发生，可直接忽略；
- 对现在这种情况而言，这个公式已经足够了，我们可以用它进行下一步了。

分析工程的重点不在于找出可以处理所有的偶然情况的复杂的解决方案。重要的是，要深化对问题的数据分析式思考，从而明确数据挖掘的作用，考虑业务约束、成本和收益，并有意识地、明确地简化假设。这样下来，项目成功的概率就会提升，部署过程中出现意外的风险也就降低了。

第12章

其他数据科学任务与技术

基本概念：作为许多数据科学常用技术基础的基本概念；熟悉数据科学构件块的重要性

示例方法：关联和共现；行为分析；链路预测；数据约简；潜在信息挖掘；电影推荐；误差的偏差－方差分解；模型集成；数据因果推理

正如前面章节中所探讨的，一种从数据分析角度考虑团队处理商业问题的有效方法，就是想象他们在处理**工程**问题——此工程不是机械工程，更不是软件工程，而是**分析工程**。商业问题本身提供了其解决方案的目标和约束条件，而数据和领域知识提供了原材料，数据科学则提供了可以将问题分解为子问题的框架以及用于解决这些问题的工具与技术。本书已经探讨过了其中一些最有价值的概念框架和一些最常用的解决方案的构件块。然而，数据科学博大精深，甚至包含了一整套学位课程，因此本书不可能面面俱到。不过幸运的是，本书所讨论的基本原则是大部分数据科学的基础。

与其他工程问题相同，把新问题分解成一系列能够处理的小问题，比从头建立一套自定义解决方案要高效得多。分析工程也没什么不同，数据科学同样会提供大量工具，以便我们处理常见或特殊的任务。因此，我们将用一些最常见的工具和方法来阐释基本原理。这些工具和方法包括查找相关性 / 找到富信息变量、寻找相似实体、分类、类概率估计、回归和聚类。

以上这些都是用于最常见的数据科学任务的工具，但第 2 章告诉我们，这样的工具还有很多。幸运的是，以上任务的内在基本概念同样也是其他任务的内在基本概念。因此，既然我们已经展示过这些概念，接下来就来简单讨论一下其他没有探讨过的任务和技术。

12.1　共现和关联：寻找匹配项

共现分组或**关联发现**是根据涉及实体的事件，找出实体间的关联。为什么要找到这样的共现呢？因为这种方法有很多应用场景。试想一个面向用户的应用场景。假设我们在做线上零售，那么根据用户的购物篮数据，我们可以告诉他："购买 eWatch 的用户也买了 eBracelet 蓝牙扬声器伴侣。"如果这些关联的确捕捉到了真实的消费者偏好，那么不只交叉销售收入可能得到提高，消费者体验也会同时得到提升（在本例中，就是使本来非立体声的 eWatch 可以播放立体声音乐）。因而这些关联充分利用了数据资产，创造了额外的客户忠诚度。

请考虑另一个业务应用场景：把产品从全球范围内的许多配送中心配送给线上客户。不是所有配送中心都有全部商品的库存，实际上，规模较小的、区域性的配送中心只会存储卖得较好的产品。尽管建立这些区域配送中心是为了降低运费，然而实际上我们会发现，仍然有很多订单要么必须从主配送中心发货，要么需要进行多次配送。原因是，在同一个订单中，客户不仅会购买人气商品，也会购买不那么受欢迎的商品。这个商业问题就可以通过挖掘数据中的关联来解决。如果一些不那么受欢迎的商品的确常常和人气商品同时出现，那么我们也可以在区域性配送中心增加它们的库存，从而大幅降低运费。

共现分组就是在数据中寻找统计数字"有趣"的数据项组合。虽然设计该任务的方法有很多种，但是请把共现当作一条规则："如果 *A* 出现，那么 *B* 也有可能出现。"那么，*A* 可能是出售 eWatch，而 *B* 可能是出售 eBracelet。[1]"有趣"的统计数据通常会遵循基本原则。

首先，我们需要控制复杂度。共现关系可能有成千上万种，其中很多只是偶然的，而不是可泛化的模式。一种控制复杂度的简单方法是建立约束条件，即符该合共现规则的数据必须达到某个最小比例，比如要求至少所有交易的 0.01% 符合该共现规则。这就叫作关联的**支持度**。

其实在关联中还有一个"可能"的概念：如果客户买了 eWatch，那么他可能也会购买 eBracelet。我们希望所找到的关联符合某个最小的可能性，并用已经见过的一些概念来量化这个概念。我们已经知道 *A* 发生时 *B* 也发生的概率是 $p(B\,|\,A)$，它在关联挖掘中被称为规则的**置信度**或**强度**。在这里还是称之为"强度"，和统计中的置信度区别开。因此，我们可以说要求规则的强度超过某个阈值，比如 5%（也就是说，在 5% 或者更多的情况下，购买 *A* 的客户同时也购买了 *B*）。

12.1.1　度量意外：提升度和杠杆率

最后，我们还是希望关联能在某种意义上让我们感到"意外"。数据挖掘中有许多关于意外的定义，但其中大部分将所发现的知识同先验背景知识、直觉和常识相联系。换句话说，关联只有在与我们之前知道或相信的事情相悖时，才算是意外的。虽然研究者们研究了如何处理这种很难编纂的知识，但是在实践中，自动处理它们并不常见。相反，数据科学家和商业用户往往会对长长的关联列表进行深入分析，从而剔除那些意料之中的关联。

注 1：*A* 和 *B* 也可以是多个数据项，比如下文中的 Facebook 点赞。但目前我们假定它们都是单个数据项。

然而，有一个度量意外的指标可以仅根据数据计算出，该指标尽管相对较弱，却很直观。它就是**提升度**，即某种关联出现的频率比偶然出现高多少，我们已经在另一个情景中遇到过它了。如果超市购物篮数据中的关联告诉我们人们经常同时购买面包和牛奶，我们可能会觉得理所应当，因为购买牛奶和购买面包的人都很多。因此我们认为它们频繁同时出现仅仅是偶然。如果关联出现的次数比偶然情况下更频繁，那么该关联就是意料之外的。提升度的计算仅需应用概率的基本概念。

公式 12-1：提升度

$$\text{提升度}(A,B)=\frac{p(A,B)}{p(A)p(B)}$$

简单地说，A 与 B 的共现的提升度是两者实际同时出现的概率，与两者不相关（互相独立）时同时出现的概率相比较的结果。和之前所学的提升度用法一样，大于 1 的提升度指 A 的出现“提升”了 B 出现的可能性。

提升度只是计算所发现的关联出现的概率比偶然情况下高多少的方法的其中一种，另一种方法则是计算两个量的差值（而非比例），被称作**杠杆率**。

公式 12-2：杠杆率

$$\text{杠杆率}(A,B)=p(B,A)-p(A)p(B)$$

你需要花点时间来理解两种方法。其中一种适用于几乎不可能偶然出现的关联，而另一种则更适用于相对更可能偶然出现的关联。

12.1.2　示例：啤酒和彩票

在“eWatch 和 eBracelet”的示例中，我们已经得知，关联发现通常用于在购物篮分析中寻找和分析所购物品的共现关系。请看另一个具体示例。

假设我们开了一家小型便利店，人们会过来购买杂货、酒、彩票等。再假设我们一年进行一次交易分析，而在这次分析中我们发现，人们常常会同时购买啤酒和彩票。但我们也知道，人们在店里购买啤酒和彩票都是常事。假设交易总量的 30% 包含啤酒，而同时包含啤酒和彩票的交易居然占 20%！这样的共现是有趣的吗？还是单纯因为两种商品太受欢迎？关联统计量可以帮我们做出判断。

首先，我们要陈述一条代表这种信念的关联规则：“购买啤酒的客户也可能购买彩票”，或更简洁地说，“啤酒 ⇒ 彩票”。然后，让我们计算这种关联的提升度。已知一个所需值：$p(\text{啤酒}) = 0.3$。假设彩票的人气也很高：$p(\text{彩票}) = 0.4$。如果两种商品完全不相关（独立），那么它们被同时购买的概率就是两者各自概率的乘积：$p(\text{啤酒}) \times p(\text{彩票}) = 0.12$。

我们还已知人们同时购买两种商品的实际概率（即数据中的频率）$p(\text{彩票}, \text{啤酒})$，该概率通过在收款机数据中寻找所有包含啤酒和彩票的交易得知。如上所述，因为 20% 的交易同时包含两者，也就是 $p(\text{彩票}, \text{啤酒}) \times 0.2$，所以提升度是 0.2/0.12，约为 1.67。这表示事实上同时购买彩票和啤酒的概率是偶然情况下同时购买二者的概率的 1.67 倍。从而，我们可以推断二者的确存在一定关系，但二者共现的主要原因还是它们都颇具人气。

那么杠杆率呢？此处杠杆率是 p(彩票 , 啤酒) − p(彩票) × p(啤酒)，也就是 0.2−0.12 = 0.08。不管出于什么原因，总之二者实际被同时购买的概率比仅仅因为二者都是人气商品而导致的同时购买概率高出了 8 个百分比。

我们还需计算另外两个重要统计量：支持度和强度。关联的**支持度**就是同时购买二者在总体数据里所占比例，即 p(彩票 , 啤酒)，其值为 20%，而**强度**则指条件概率，即 p(彩票 | 啤酒)，其值为 67%。

12.1.3　Facebook点赞的关联

虽然关联发现通常用于购物篮数据，有时甚至被称作**购物篮分析**，但是这个技术的应用其实比这更为普遍。我们可以用第 9 章的 Facebook“点赞”示例来说明。回忆一下，我们拥有大量关于 Facebook 用户“点赞”过的事物的数据（Kosinski, Stillwell & Graepel, 2013）。类比购物篮数据，可以认为每个用户都有一个“点赞”的篮子，装有该用户所有的点赞数据。现在请回答这个问题：某些“赞”实际同时出现的情况是否比偶然同时出现的情况更频繁？虽然这个有趣的示例将仅用于说明关联发现，但整个过程其实有重要的商业应用。如果你是一个希望了解某个特定市场中客户的销售人员，那你可能会想要找到人们点赞的模式。如果你从数据分析角度进行思考，那么你将恰好应用本章中我们到目前为止说明过的思维方式：你会想要知道哪些内容的共现比偶然情况下更频繁。

在开始挖掘数据前，先介绍一个更有助于关联发现的概念。由于我们现在使用购物篮作为类比，因而要考虑物品所指代的究竟是什么。为什么不把所有可以用于寻找我们感兴趣的关联的数据都放进篮子？比如，我们可以把用户定位放进篮子，然后观察点赞和定位间的关联。在实际购物篮数据中，这些项有时被称作**虚拟项**，以便将其与人们逛商店时实际放进购物篮的物品区别开来。关于 Facebook 数据，我们曾经获取了许多用户的心理测量数据，比如外向或随和的程度、IQ 测试的得分等。通过关联发现来寻找这些心理测量特征间的关联，应该会很有趣。



有监督和无监督

我们必须谨记有监督型和无监督型数据挖掘的区别。如果想要具体理解与随和的性格或与给我们的品牌点赞最相关的数据项，我们就应该构建一个有监督的问题，附带对应的目标变量。这正是第 9 章中的证据提升度和本书中所有有监督划分所做的。如果在没有具体目标的情况下探索数据，那么关联分析会更加适用。有监督和无监督挖掘的区别可以参考第 6 章在聚类背景下的讨论，而那些基本概念也适用于关联挖掘。

好了，接下来我们看看 Facebook 的点赞数据中究竟存在什么关联。[2] 寻找这些关联用到了一个常用的关联挖掘系统 Magnum Opus[3]，它能寻找提供最大提升度或最高杠杆率的关联，同时过滤因出现次数太少而不够有趣的关联。下文中的列表展示了 Facebook 点赞中一些提升度最高的关联，此处关联的阈值是至少包含数据集中 1% 的用户。这些关联是否有意

注 2：感谢 Wally Wang 的鼎力相助。

注 3：详见 http://www.giwebb.com/。

义？是否能告诉我们用户喜好之间的关系？你会发现这些提升度都大于 20，也就是说，这些关联至少比偶然出现的**概率高 20 倍**。

```
《恶搞之家》 & 《每日秀》 -> 《科尔伯特报告》
支持度=0.010；强度=0.793；提升度=31.32；杠杆率=0.0099

《千与千寻》 -> 《哈尔的移动城堡》
支持度=0.011；强度=0.556；提升度=30.57；杠杆率=0.0108

Selena Gomez -> Demi Lovato
支持度=0.010；强度=0.419；提升度=27.59；杠杆率=0.0100

I really hate slow computers & Radom laughter when remembering something ->
    Finding Money In Your Pocket
支持度=0.010；强度=0.726；提升度=25.80；杠杆率=0.0099

彩虹糖 & 荧光棒 -> high起来!
支持度=0.011；强度=0.529；提升度=25.53；杠杆率=0.0106

Linkin Park & Distrubed & System of a Down & Korn -> Slipkont
支持度=0.011；强度=0.862；提升度=25.50；杠杆率=0.0107

Lil Wayne & Rihanna -> Drake
支持度=0.011；强度=0.619；提升度=25.33；杠杆率=0.0104

彩虹糖 & 激浪 -> 佳得乐
支持度=0.010；强度=0.519；提升度=25.23；杠杆率=0.0100

海绵宝宝 & 匡威 -> 派大星
支持度=0.010；强度=0.654；提升度=24.94；杠杆率=0.0097

Rihanna & Taylor Swift -> Miley Cyrus
支持度=0.010；强度=0.490；提升度=24.90；杠杆率=0.0100

Disturbed & Three Days Grace -> Breaking Benjamin
支持度=0.012；强度=0.701；提升度=24.64；杠杆率=0.0117

Eminem & Lil Wayne -> Drake
支持度=0.014；强度=0.594；提升度=24.30；杠杆率=0.0131

Adam Sandler & System of a Down & Korn -> Slipknot
支持度=0.010；强度=0.819；提升度=24.23；杠杆率=0.0097

Pink Floyd & Slipknot & System of a Down -> Korn
支持度=0.010；强度=0.810；提升度=24.05；杠杆率=0.0097

音乐 & 日本动画 -> 日本漫画
支持度=0.011；强度=0.675；提升度=23.99；杠杆率=0.0110

中等IQ & 酸味条状凝胶糖果 -> I Love Cookie Dough
支持度=0.012；强度=0.568；提升度=23.86；杠杆率=0.0118

Rihanna & Drake -> Lil Wayne
支持度=0.011；强度=0.849；提升度=23.55；杠杆率=0.0104

I Love Cookie Dough -> 酸味条状凝胶糖果
支持度=0.014；强度=0.569；提升度=23.28；杠杆率=0.0130
```

```
Laughing until it hurts and you can't breathe! &  I really hate slow computers ->
    Finding Money In Your Pocket
支持度=0.010；强度=0.651；提升度=23.12；杠杆率=0.0098

Evanescence & Three Days Grace -> Breaking Benjamin
支持度=0.012；强度=0.656；提升度=23.06；杠杆率=0.0117

迪士尼 & 迪士尼乐园 -> 迪士尼世界
支持度=0.011；强度=0.615；提升度=22.95；杠杆率=0.0103

i finally stop laughing... look back over at you and start all over again ->
    That awkward moment when you glace at someone starting at you.
支持度=0.011；强度=0.451；提升度=22.92；杠杆率=0.0104

Selena Gomez -> Miley Cyrus
支持度=0.011；强度=0.443；提升度=22.54；杠杆率=0.0105

锐滋花生巧克力 & 星爆果汁软糖 -> 家乐氏果酱吐司饼干
支持度=0.011；强度=0.493；提升度=22.52；杠杆率=0.0102

彩虹糖 & 海绵宝宝 -> 派大星
支持度=0.012；强度=0.590；提升度=22.49；杠杆率=0.0112

迪士尼 & 多莉[4] & 《玩具总动员》 -> 《海底总动员》
支持度=0.011；强度=0.777；提升度=22.47；杠杆率=0.0104

Katy Perry & Taylor Swift -> Miley Cyrus
支持度=0.011；强度=0.441；提升度=22.43；杠杆率=0.0101

AKON & Black Eyed Peas -> Usher
支持度=0.010；强度=0.731；提升度=22.42；杠杆率=0.0097

Eminem & Drake -> Lil Wayne
支持度=0.014；强度=0.807；提升度=22.39；杠杆率=0.0131
```

大部分应用关联挖掘示例的领域（如 Facebook 点赞），读者比较了解。这是因为，由于挖掘是无监督的，因而在评估环节更关键的是领域知识验证（回忆第 6 章的探讨）；而如果不这么选择，那么目标任务的定义就可能因不够明确而不能用于客观评估。然而，关联挖掘的一种有趣的实际用途就是研究我们不那么了解的数据。假设你要开始一项新工作，探索公司客户的交易数据以检验强共现关系可以让你很快对客户群的喜好有一个大体认识。因此，考虑到这一点，请回顾 Facebook 点赞中存在的共现，但是假装这并不是流行文化领域：这些关联和其他类似的关联（有大量这样的关联）会给你提供一个非常广泛的视角来了解客户喜好。

12.2　用户画像：寻找典型行为

用户画像的目的是描述个人、群组或总体的典型行为特征。比如有这样一个问题："这个客户分组的典型信用卡使用习惯是什么？"虽然我们可以简单地用开支的均值来进行表述，但在我们的商业问题中，这样简单的描述可能无法充分代表用户行为。举个例

注 4：《海底总动员》角色。——译者注

子，欺诈检测经常会使用用户画像来描绘用户的一般行为，然后找出用户明显不符合常态的行为，尤其是那些曾经指示出欺诈现象的行为（Fawcett & Provost, 1997; Bolton & Hand, 2002）。欺诈检测的画像可能需要关于用户工作日和周末信用卡使用均值、信用卡跨国使用情况、商家和产品类别的使用情况和可疑商家的使用情况复杂的描述。用户行为的画像一般可以在整个总体、小群体、甚至每个人的层面上来进行。比如，针对每个信用卡用户，我们可以根据其信用卡的跨国使用情况来画像，以免因为出国旅游而产生过多误报。

用户画像对前面探讨过的概念进行了组合，如果总体中存在不同行为的子组，那么用户画像也会对其进行聚类。许多画像方法看似复杂，实际上却只是第 4 章介绍的基本概念的体现：先用一些参数定义一个数值函数，再定义一个目标，然后找出最符合目标的参数。

那么请考虑一个企业运营管理的简单例子。企业想通过数据得知自己的客户服务中心为客户提供个人支持的效果如何，[5] 好的个人支持的其中一个方面是不让客户长时间等待。那么，我们该如何描绘打电话到客服中心的客户的一般等待时间呢？答案是计算等待时间的均值和标准差。

这种做法似乎正是略懂基础统计的管理者会做的，实际上这也是模型拟合的一个简单例子，其原因如下。假设客户的等待时间服从正态分布（也称高斯分布，这样的说法可能会让不懂数学的读者望而却步，但这其实只代表该分布是一条钟形曲线，且有许多优良性质）。重要的是，这条曲线是等待时间的“画像”，（本例中）只有两个重要参数：均值和标准差。一旦计算出两者，我们就找到了在正态分布的假设下，描述等待时间的“最佳”画像或模型。本例中的“最佳”和逻辑回归里的“最佳”含义相同，比如，根据开支所计算出的均值可以告诉我们最可能生成该数据（“最大似然”模型）的高斯分布的均值。

这个观点说明了，为什么数据科学视角在简单情景下也会有帮助：在计算平均值和标准差时，尽管对学过的统计知识印象已经很模糊了，但我们对正在做的事情比之前清楚得多。我们还需要牢记在第 4 章介绍并在第 7 章详细阐述的基本原则：要想清楚我们从数据科学结果中究竟要得到什么。而这里我们想描绘的是客户的“一般”等待时间。如果根据绘图的结果，数据看起来不像高斯分布（对称钟形曲线，在尾部迅速降为 0），那我们就可能需要考虑计算均值和标准差了，我们也可以计算中位数（因为中位数对偏度不敏感）或去拟合另一种分布（可能需要和从事统计学的数据科学家讨论一下哪种更合适）。

为说明精通数据科学的管理者可能如何继续处理该问题，我们来看一下几个月内客户给银行客服中心致电的等待时间的分布，见图 12-1。重要的是，我们能看出对分布的可视化如何帮助我们发现数据科学上的问题。因为图中的分布并不是对称的钟形曲线，所以接下来，我们认为单纯通过均值和标准差来描述等待时间不太合理。比如，均值（100）似乎并不能描述客户的一般等待时间，因为这个值太大了。技术上，因为分布的“长尾”会导致均值偏高，所以这并不能真实反映大部分数据的实际位置，因而不能真实反映客户的一般等待时间。

为了更深入地了解精通数据科学的管理者处理该问题的方法，本章将对此做进一步探讨，但只会探讨处理偏态数据的一种常用方法，不会深入细节：对等待时间取对数（log）。

注 5：欢迎感兴趣的读者阅读 Brown 等人在这方面的技术处理和细节（2005）。

图 12-2 中所展示的分布与图 12-1 相同，但对等待时间取了对数。我们可以看到，在进行了一小步转化之后，等待时间的分布就很像经典钟形曲线了。

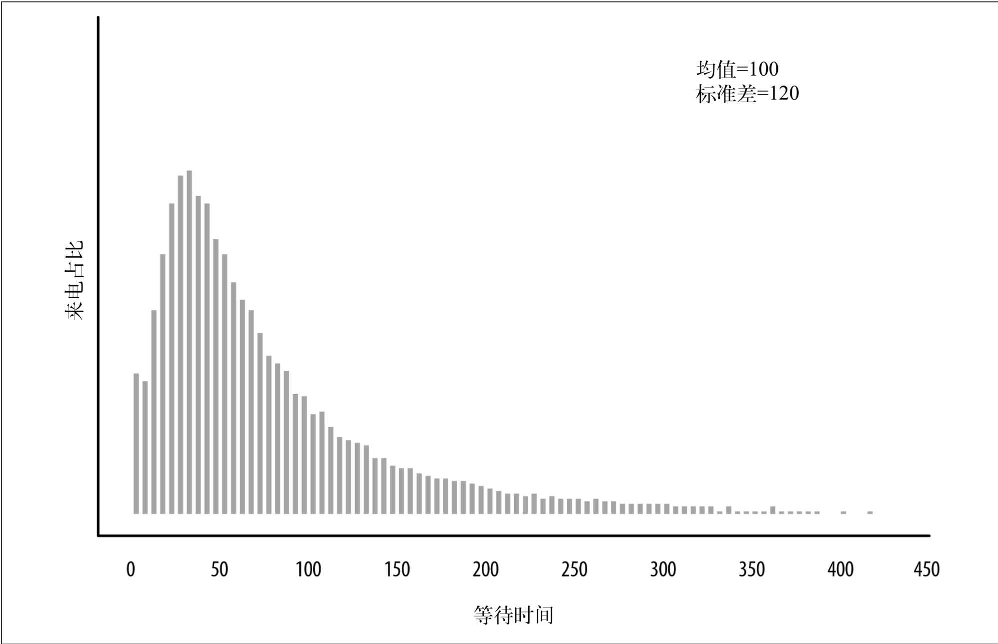


图 12-1：客户给银行客服中心致电的等待时间的分布

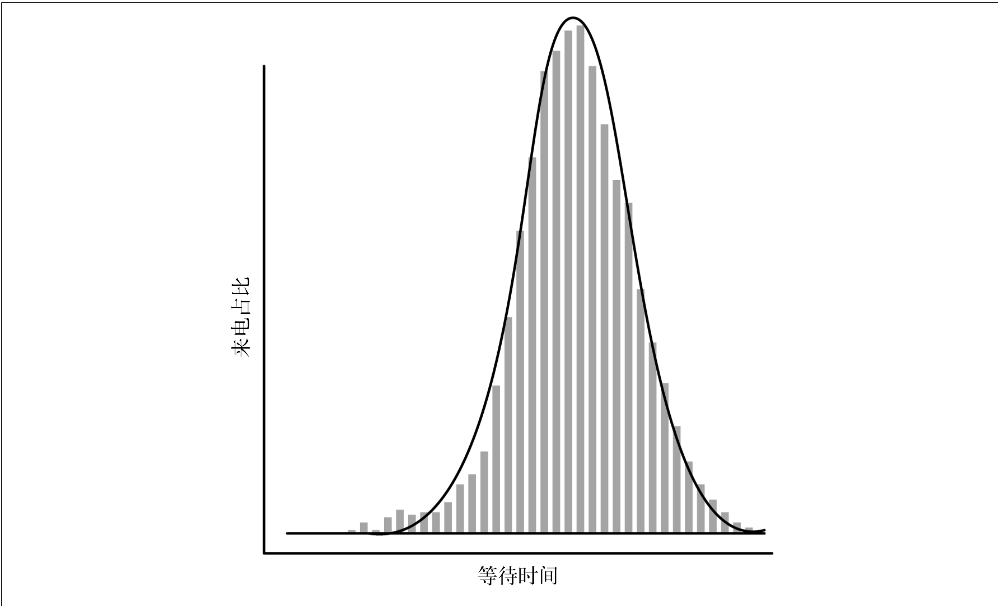


图 12-2：对数据略作重新定义后，客户给银行客服中心致电的等待时间的分布

实际上，如上文所述，图 12-2 还展示了符合钟形分布的高斯分布（钟形曲线）。它确实拟合得很好，因此我们可以将均值和标准差作为取对数后等待时间的概括统计量。[6]

这个简单的示例可以很好地扩展到更复杂的情形中。换个背景，假设要根据用户在我们网站上所花的钱和时间对用户行为进行画像。如图 12-3 中的数据点所示，我们认为这两者相关，但不完全相关。在这里要重申一种非常常用的方法，也就是第 4 章中所学的基本概念：选定一个参数化数值函数和一个目标，然后寻找使目标达到最大值的参数。比如，我们可以选择二维高斯函数，它的图像并不是一条钟形曲线，而是一个钟形**椭圆**，即一个中心密度极大、越靠近边缘密度越小的椭圆形斑点。图 12-3 将该图像表现为等高线形式。

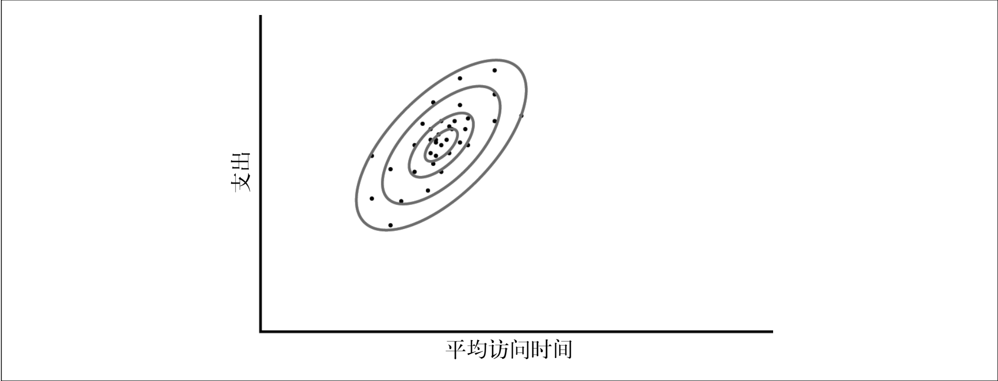


图 12-3：根据用户在我们网站上所花费的钱和时间所构造的用户画像，以数据的二维高斯拟合的形式表示

我们可以继续提升用户画像的复杂程度。如果我们认为客户群中存在不同的子组，而不同子组中的客户行为不同，那么该怎么办呢？可能我们就不会愿意只用高斯分布来拟合客户行为了。然而，我们可能乐于假设客户被分为 k 组，每个组的行为都服从正态分布，然后用多个高斯函数来拟合模型。我们把这样的模型称为**高斯混合模型**（GMM）。然后我们再次应用基本概念，用最大似然参数找出最符合数据的 k 维高斯函数（根据具体的目标函数而定）。图 12-4 中 $k = 2$，该图像展现了拟合过程识别出客户中 2 个群体的过程，其中每个群体都用二维高斯分布刻画。

注 6：接受过统计训练的数据科学家可能一眼就能看出原始数据的分布形状，如图 12-1 所示。这就是所谓的对数 − 正态分布，即问题中的量的对数形式呈正态分布。

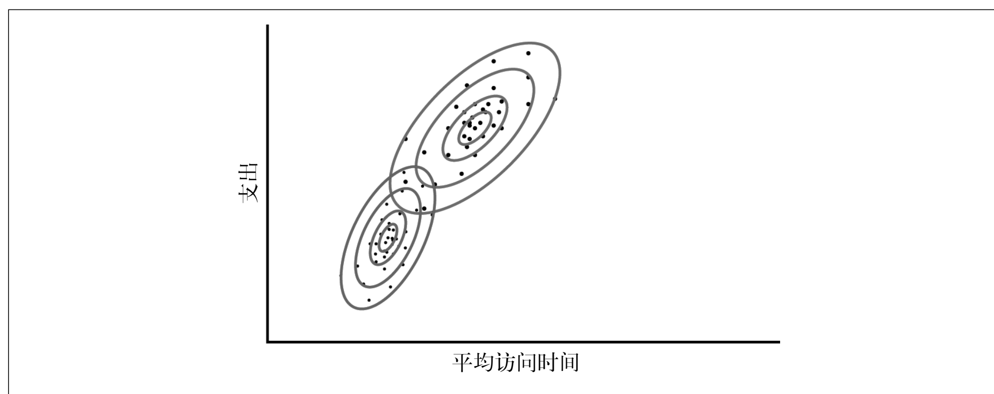


图 12-4：根据客户在我们网站上所花的时间和钱，对客户构建用户画像，并用高斯混合模型（GMM）表示，其中拟合数据的高斯函数均为二维。GMM 在这两个维度上对客户进行了“软”聚类

现在我们有了相当复杂的用户画像，这是对基本原则的一次简单应用。需要注意的一点是，虽然 GMM 能进行聚类，但它与第 6 章中的聚类方法不同。本例展示了基本概念（而不是某个任务或算法）是如何构成数据科学的基础的。在此情况下，聚类像分类和回归一样，可以有很多途径。



“软”聚类

顺便一提，你可能发现 GMM 生成的簇会互相重叠，这是因为 GMM 进行的是“软”聚类，也称概率聚类。其中，每个点并不严格地属于某个簇，而是被赋予属于每个簇的程度或概率。在这样的聚类中，尽管我们可以认为某些点（比起其他簇）更有可能来自某个簇，然而这些点仍有可能来自任何簇。

12.3 链路预测和社交推荐

有时，与其预测数据项的性质（目标变量值），不如预测数据项之间的**关系**。一个常见例子就是预测两个人之间的关系。链路预测在社交网络系统中非常常用，比如：“既然你和 Karen 有 10 个共同好友，那么或许你想关注一下 Karen？”链路预测也能估计关系的强度。比如，在给用户推荐电影时，我们可以把用户和他们所看过或评过分的电影设想成一幅图，并在其中寻找那些不存在但根据预测应该存在且强度较大的关系，而这些关系就是推荐的基础。

链路预测的方法有很多，即使本书用一整章篇幅也无法详尽描述。然而，我们可以根据数据科学的基本概念理解许多不同方法。请考虑一个社交网络的案例：如果要预测两个人之间是否存在关系或预测该关系的强度，你将如何根据当前所学知识定义问题？这里有一些选择。我们可以假设存在关系的个体之间存在相似性。然后，基于应用场景中的重要方面，我们需要定义一个相似性度量。

我们是否能在两个人之间定义一个相似性度量，来度量他们是否可能想成为朋友（或者已经成为好友，视情况而定）？当然可以。直接使用上面的例子，我们可以将相似性视为共享朋友的数量。当然，相似性度量应该更加复杂。首先根据与好友的互动量、地理邻近程度或其他因素，对好友进行加权处理，然后找出或设计出包含这些因素的相似性函数。我们可以把这种好友强度作为相似度的一方面；而相似性同时也包含其他方面（因为在学习过第 6 章后，我们更习惯于使用多变量相似性），比如相同的爱好、相同的人口统计学资料等。本质上，我们可以通过思考把人表示为数据的不同方法来对人应用“寻找相似数据项”的知识。

这是处理链路预测问题的一种方法。接下来，请再考虑另外一种，以说明这些基本原则是如何应用到其他任务中的。由于我们想**预测**链路的存在性（或强度），因而可能需要把问题定位为预测建模问题，因此就要应用到预测建模问题的思考框架。和以前一样，先进行业务理解和数据理解。什么是数据项？一开始我们可能觉得所要关注的是**两个**实例之间的**关系**。因此概念框架就派上用场了：保持我们的一贯做法，定义一个要预测的实例。那么我们想预测的到底是什么？是两人之间关系的存在性（或强度，但现在先只考虑存在性），因此一个实例应为两个人！

一旦我们把一个实例定义为两个人，就可以顺利往下进行了。下一步，其目标变量是什么？是关系是否存在，或一旦进行推荐，是否能形成关系。这是一个有监督问题吗？是的，我们可以获取到链路已经存在或不存在情况下的训练数据。如果我们想更谨慎，那么也可以进行投资以获取专门用于推荐问题的标签数据（可能需要比定义关系的确切语义花更多的时间）。其特征是什么？其特征是**这两个人**的特征，比如他们有多少共同好友、爱好有多相似等。既然我们已经把问题定位为预测建模问题，那么就可以开始寻找合适的模型和评估模型的方法了。这和一般预测建模问题经历的概念过程相同。

12.4　数据约简、潜在信息和电影推荐

针对某些商业问题，我们希望把大型数据集替换成较小的数据集，但是该小数据集要保留大数据集中的大部分重要信息。较小的数据集不仅处理起来更方便，或许还能更好地展现其中的信息。比如，消费者观影偏好的大数据集可以简化成小数据集，并且展示出观影数据中隐藏的消费者品味偏好（比如对电影类型的偏好）。虽然这样的数据约简通常需要牺牲一些信息，但是在数据的洞察或易处理性与信息损失之间进行权衡非常重要，而这种权衡往往证明牺牲信息是值得的。

数据约简和链路预测一样，都是一种一般任务，而不是一种特殊技术。技术有很多种，可以通过基本原则来了解。让我们以一种常用技术为例来进行讨论。

继续讨论电影推荐问题。电影租赁公司 Netflix ™出资百万美元举办了一场如今（至少在数据圈内）非常知名的比赛，以角逐出能最好地预测用户对电影的评分的个人或团队。Netflix 在保留数据集上定义了一个预测效果目标，首先达到该目标的参赛者将会获得奖励。[7] 参赛

注 7：Netflix 挑战的规则包含许多技术细节，你可以在维基百科相关网页中阅读。

者可以利用 Netflix 提供的用户电影评级历史数据。虽然获胜组[8]构造了一种极其复杂的技术，但他们的成功主要归功于其解决方案的两个方面：其一，使用了集成模型，这一点后面 12.5 节将进行探讨；其二，数据约简。我们可以轻而易举地用基本概念来描述他们主要使用数据约简技术。

这个待解决的问题实质上是一个链路预测问题，要预测的具体内容是用户和电影之间的链路强度，该强度代表用户有多喜欢这部电影。我们刚刚探讨过，这个问题可以被定位为预测建模问题，那么，用户和电影之间关系的特征是什么呢？

一种最常用方法是把模型建立在偏好的**潜在**维度的基础上。Netflix 这场比赛的许多获胜者在其共同撰写的论文中对该方法进行了详细描述（Koren, Bell & Volinsky, 2009）。"潜在"一词在数据科学中指的是"相关，但在数据中不明显"。第 10 章探讨的主题模型，是潜在模型的另一种形式，其中潜在信息指的是一系列文档主题。在这里，电影偏好的潜在维度包括了可能的特征，比如是严肃的还是逃避现实的、是喜剧片还是剧情片是否面向儿童，以及性别取向。即使这些特征没有明显出现在数据中，也会对用户对该电影的喜好造成巨大影响。由于潜在维度将会从数据中浮现，因而这些维度可能还包括一些难以明确定义的内容，如人物的深度或情节离奇程度，以及一些从未被明确表述过的维度。

再次重申，我们可以把这种数据科学高级方法认作基本概念的组合。用潜在维度进行电影推荐的思路是，把每部电影用潜在维度表示成特征向量，同时把每个用户的偏好也用潜在维度表示成特征向量，然后计算用户和所有电影的相似度评分，并据此向用户推荐电影。因为当两者都用潜在维度表示时，最符合用户偏好的电影就是与用户最相似的电影。

图 12-5 展示了一个根据 Netflix 电影数据挖掘出的二维潜在空间[9]，以及在这个空间中展示的一系列电影集合。要解释这些从数据中挖掘出的潜在维度，必须依赖于数据科学家或商业用户的推断，最常用的方法是观察这些维度如何分离电影，然后把领域知识应用其中。

注 8：优胜组"Bellkor's Pragmatic Chaos"包含 7 名成员。大赛的历史和团队的发展历程十分复杂有趣，你可访问 Netflix 大奖的维基百科网页获取更多信息。

注 9：感谢获奖组成员之一 Chris Volinsky 的帮助。

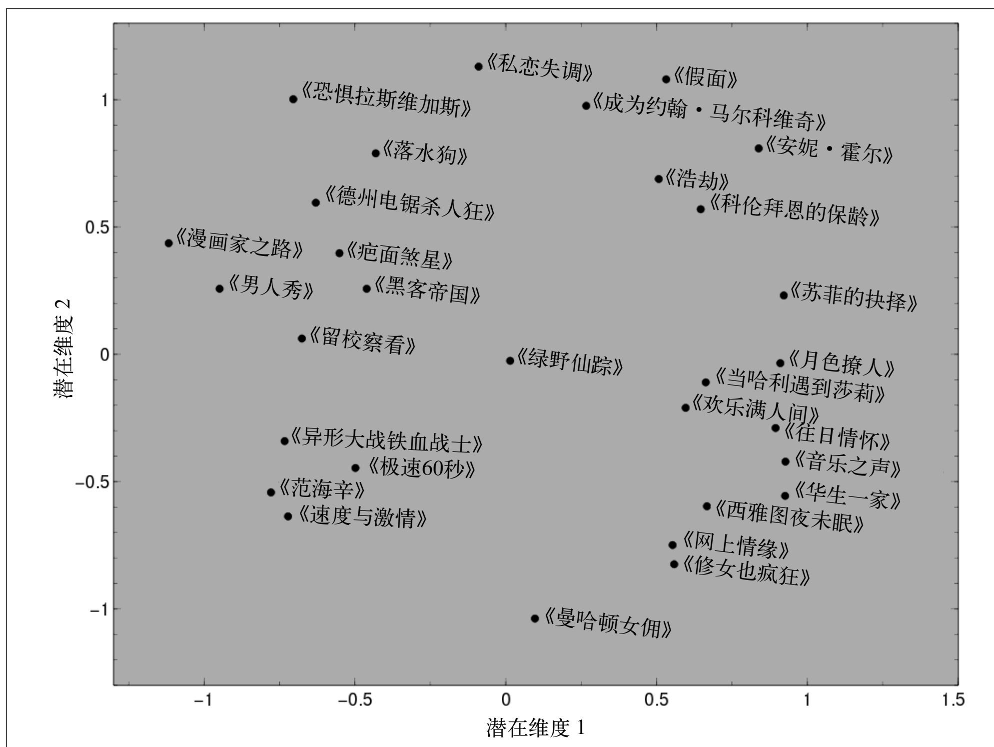


图 12-5："品味空间"中用 Netflix 挑战数据中挖掘出的两种最强的潜在维度定义的一系列电影。下文包含详细探讨。根据其观看过或评过分的电影，也可以将用户绘进这个空间。基于相似性的推荐方法会像候选人推荐一样，把与用户距离最近的电影推荐给用户

图 12-5 中，横轴代表的潜在维度似乎能把电影分成右侧的剧情片和左侧的动作片，在轴的两端，最右侧的是走心电影，比如《音乐之声》《月色撩人》和《当哈利遇到莎莉》，而最左侧的电影与走心电影相反（走胆电影？），包含男人和青少年的刻板形象（《男人秀》《留校察看》）、杀戮（《德州电锯杀人狂》《落水狗》）、速度（《速度与激情》）和打怪（《范海辛》）。纵轴代表的潜在维度则似乎把电影分成了知性诉求型和情感诉求型，一端包含《成为约翰·马尔科维奇》《恐惧拉斯维加斯》和《安妮·霍尔》，而另一端则包含《曼哈顿女佣》《速度与激情》和《网上情缘》。你可以不同意我们对维度的解读，因为这些解读完全是主观的。但有件事是清晰的：《绿野仙踪》在潜在维度代表的几种品味中做到了不正常的平衡。

为使用该潜在空间进行电影推荐，我们必须根据用户租赁过或评价过的电影，把用户也放进这个空间里，这样一来，与用户所在位置最接近的电影就是最适合推荐给该用户的电影。注意，在进行推荐时，必须要不断回忆对业务的理解。比如，不同的电影有不同的利润空间，而我们可能希望将这些知识与最相似电影的知识结合起来。

不过，我们如何在数据中**找到**正确的潜在维度呢？应该应用第 4 章介绍的基本概念，把用户和电影间的相似度计算表示成数学公式的形式，并用字母 d 来代表仍然未知的潜在维度。每

个维度都应表示成每部电影和每个用户的一系列权重（系数）。权重越高，说明该维度与电影或用户的联系越强。维度的含义应完全隐含在电影或用户的权重中。比如，我们在看到某些维度上权重很高的电影和权重很低的电影后，可能会认为"高分电影都很'离奇'"。这种情况下，我们就可以把维度想成是电影的**情节离奇程度**，但请谨记，这种对维度的解读是我们强加上的，维度其实只是电影根据用户评分在数据中形成的某种聚类形式。

回想一下，为了让数值函数模型拟合数据，我们要找到数值函数的一系列最优参数。最初，维度 d 只是数学上的抽象表示。仅当拟合数据的参数选好后，我们才能定义潜在维度的含义（有时会徒劳无功）。在这里，函数的参数应该是每个用户和每部电影在潜在维度上的（未知）权重。直观地说，数据挖掘需要同时判断电影情节离奇程度和该观影者有多喜欢情节离奇的电影。

现在还需要一个目标函数来判断拟合优度。我们根据已观察到的电影评分数据来定义用于训练的目标函数，并在这些维度上找到一系列描述这些用户和电影的权重。其实，许多目标函数都可以用于电影推荐问题。比如，我们可以选择最能预测训练集中观测评分的权重（正则化，如第 4 章所讨论的）。或者，我们可以选择最能解释所观测的评分变动的维度。这种方法常被称为"矩阵分解"，感兴趣的读者不妨从关于 Netflix 挑战的论文（Koren, Bell & Volinsky, 2009）开始读起。

结果是，在简化了的维度集上，我们对每部电影都做了表示（可能是情节离奇程度、是否为一部"催泪电影"或"针对男性的电影"等）这些维度是根据训练集找出的 d 个最佳潜在维度。现在可以回头看看图 12-5 和相关讨论，其中包含两个最能拟合数据的潜在维度，也就是用 $d = 2$ 的二维变量拟合数据选出的最佳维度。

12.5　偏差、方差和集成方法

Netflix 大赛的优胜者还使用了另一种常用的数据科学技术：构造很多推荐模型，并把它们组合成一个"超级模型"。用数据挖掘术语来说，即他们构造了一个**集成**模型。据观察，在很多情况下，集成模型可以提高模型的泛化能力。这不仅适用于推荐问题，还广泛适用于分类、回归、类概率估计等问题。

为什么模型的组合往往优于单个模型呢？如果我们把每个模型当作目标预测任务中的一种"专家"，那么模型的组合就是一群专家。与其只咨询一位专家，不如请教一群专家然后将他们的建议进行组合。比如，我们可以让他们对分类结果投票，或对他们的数值型预测取平均。注意，这是第 6 章介绍的将相似性计算转化成"最近邻"预测模型方法的拓展。在进行 *k*- 最近邻预测时，我们要找到一组相似示例（即一些非常初级的专家），然后根据一些函数来组合它们的预测结果。因此 *k*- 最近邻模型就是一种简单的集成模型。通常，集成模型会把更复杂的预测模型作为自己的"专家"。比如，它可能会构造一组分类树，然后把预测值的均值（或加权均值）作为结果。

集成模型在什么情况下会提高模型效果呢？当然，如果每个专家知道的事情完全**一样**，那么他们就会给出相同的预测，这会使集成模型的优势无法体现；而如果每个专家理解问题的角度稍有不同，那么他们就会给出互补的预测，因而整个专家组的预测会比个人的预测信息量更大。从技术上说，我们希望专家们产生不同种类的**误差**——这些误差越不相关越

好，最好能完全独立。在整合所有预测时，这些误差就能互相抵消，使预测真正互补，从而使集成模型优于任意一个独立的模型。



集成方法不但历史悠久，而且是数据科学研究的一个活跃领域。关于这方面的文章还有很多，感兴趣的读者不妨从 Dietterich（2000）的评论文章读起。

一种可以帮助我们理解为什么集成会有效的方法，是首先理解模型的误差可以由以下三个因素描述：

(1) 内在随机性；

(2) 偏差；

(3) 方差。

第一条，内在随机性，仅包括预测非"确定性"的情况（即我们每当看到同一组特征时，不会总是得到具有相同值的目标变量）。举个例子，特征相同的客户可能不会一直购买产品或一直不购买产品，根据现有的信息，预测可能仅是固有的概率。因此，预测中观察到的一部分"误差"仅仅是问题的固有概率性质导致的。我们可以讨论一个特定的数据生成过程是真正具有概率性还是我们根本没有看到所有必要信息。但这种争论主要是学术层面上的[10]，因为基于现有的数据，这个过程可能本质上就是概率性的。进一步假设我们已经尽可能降低了随机性，并且针对这个问题我们可以实现理论上的某个最大准确度。这种准确度就叫作**贝叶斯率**，它通常是未知的。在本节的剩余篇幅中，我们将认为贝叶斯率是"完美"的准确率。

除了内在随机性，模型还有两个产生误差的原因。首先，建模过程可能是"有偏差的"。你可以借助学习曲线（回忆 5.8 节）来理解这个概念。具体地说，如果不论用多少训练数据来训练模型，学习曲线也永远达不到完美准确率（贝叶斯率），那么建模过程就是有偏差的。比如，我们学习得到了一个用于预测某广告活动的响应情况的（线性）逻辑回归模型。如果实际的响应情况真的比模型所能表现的更复杂，那么该模型将永远无法达到完美准确率。

其次我们没有无限多的训练数据，只有一些有限的样本。建模过程通常会由于样本的细微差别而得出不同模型，而这些不同的模型的准确率也不相同。至于不同训练集（假设规模相当）导致的模型准确率的差别究竟有多少，要依建模过程的方差而定。其他条件不变时，方差更大的建模过程产生的模型误差可能更大。

你现在可能明白了，我们希望建模过程既没有偏差也没有方差，或至少偏差和方差都较小。但遗憾（且直观）的是，这两者之间一般需要权衡。方差小的模型通常偏差较大，反之亦然。举一个非常简单的例子，我们想在忽略所有客户特征的情况下，简单估计广告活动的响应情况并简单预测（平均）购买率。但是，如果客户的购买倾向存在差异，就无法得到完美准确率的模型。另一方面，我们也可能根据一千个详细变量对客户进行建模。我

注 10：这一争论有时也会取得成果，比如，通过考虑是否有所有必要信息，可能会发现我们需要获取一个新属性，从而提升预测能力。

们现在可能有机会获得更好的准确性，但我们也可以预期，基于存在略微差异的训练集所获得的模型会有更大的差异。因此，我们不一定期望一千个变量的模型更好。我们并不能确切地知道是哪个变量的误差（偏差或方差）占了主导地位。

你可能会想："当然了，我们在第 5 章学过，变量过多的模型会出现过拟合。我们应该在一定程度上对模型进行复杂度控制，比如选择一部分变量来建模。"这完全没问题，复杂度更高的模型偏差会更小，但方差会更大。复杂度控制通常试图权衡偏差和方差（一般是未知的），以找到使两者产生的误差组合最小的"甜蜜点"。因此，我们可以对一千个变量的问题应用变量选择。如果购买率的确因客户而异，而我们又有足够的训练数据，那么变量选择很可能不会把所有变量都删除，否则我们就只能对总体取平均了。我们希望能用一部分变量来建模，从而根据手头的训练数据，尽可能精确地进行预测。



技术上，本节讨论的准确率是模型准确率的期望值。我们没有指明这一点，否则讨论就会在技术上变得复杂。对偏差、方差和两者之间权衡感兴趣的读者，不妨从 Friedman（1997）的一篇技术性强但非常易读的论文读起。

现在我们知道了集成技术为什么会起作用。如果我们的建模方法方差极高，那么对多次预测取平均就能降低预测的方差。确实，集成方法会大幅度提升高方差方法的预测能力，比如在可能会出现严重的过拟合现象时（Perlich, Provost & Simonoff, 2003）。集成方法通常用于树型归纳，因为分类树和回归树往往方差较高。在集成方法领域中，你可能还会听到随机森林、套袋法和自助法，这些都是用于树形模型的常见集成方法（后两者更通用）。读者可以访问维基百科来了解有关它们的更多信息。

12.6　数据驱动的因果解释和一个病毒式营销示例

本书（第 2 章和第 11 章）提及的一个重要主题是数据的**因果解释**。预测建模对很多商业问题来说都非常有用，但目前为止本书所讨论的预测建模都基于相关关系，而不是因果关系。通常我们想更深入地研究某现象，以了解是什么影响了什么。这样做的原因可能是为了更了解我们的业务，也可能是希望用数据改进决策，以取得理想结果。

请考虑一个详细示例。最近"病毒式"营销取得了广泛关注。对"病毒式"营销的一种常见解释是消费者可以互相影响着购买产品，因此营销者可以通过对一些消费者"播种"（比如向他们提供免费产品）而大量获益。这些消费者就是"影响者"，他们会提高所认识的人购买产品的可能性。病毒式营销的目标是构造像传染病一样迅速流行的活动，但这种流行背后的关键假设是消费者之间会互相影响。那么这种影响有多大呢？数据科学家们会观测消费者获得产品后，其社交网络中的邻居购买该产品的可能性是否确实提升，并根据观测数据来度量这种影响。

然而，对数据的简单分析可能具有严重的误导性。这基于一个重要的社会学因素（McPherson, Smith-Lovin & Cook, 2001）：在社交网络中，人们倾向于结识与其相似的人。那么这个因素为什么如此重要？

因为这表示社交网络中的邻居可能会有相同的产品偏好，而由此我们可以**预期**，即使消费

者之间**不存在任何因果关系影响**，选择或喜欢这些产品的人的邻居，也会选择或喜欢该产品！的确，《美国国家科学院院刊》中写到，根据对因果分析的谨慎应用，传统的估计方式将病毒式营销的影响至少高估了 700%！

谨慎地对数据进行因果解释的方法有很多，而且都可以用一个通用的数据科学框架来理解。本书讨论这一点的目的在于，理解这些复杂的技术需要先掌握目前介绍的基本原则。谨慎的因果数据分析要求理解获取数据所做的投资，这些数据包括相似性度量、期望值计算、寻找相关关系和富信息变量、用公式拟合数据等。

第 11 章对这种更复杂的因果分析进行了一些介绍。当时我们回顾了电信公司的用户流失问题，并提出了"是否应该把最容易受到特殊优惠影响的用户作为目标"的问题。该示例说明了期望值框架所起的作用，同时也介绍了许多其他概念。在因果关系理解中，使用相似性匹配（第 6 章）模拟得到或未得到"处理"（如促使留下的激励）的"反事实情况"的技术有很多。其他因果分析方法也能用数值函数拟合数据和解读函数的系数。[11]

重点是，我们无法在不理解基本原则的前提下理解因果数据科学。因果数据分析只是一个例子，你还会在其他更复杂的方法中遇到类似技术。

12.7 小结

尽管数据科学中有许多特殊技术，然而为了透彻地了解该领域，我们需要先抛开这些技术，转而考虑应用这些技术的问题。本书关注的是一些最常见的问题（寻找相关关系和富信息变量、寻找相似数据项、分类、概率估计、回归和聚类等），并表明了数据科学的概念能为理解问题和其解决方法提供坚实基础。本章展示了另外一些重要的数据科学任务和技术，并说明通过基本概念提供的基础，这两者也可以得到很好的理解。

本章具体讨论了：寻找数据项之间有趣的共现关系或关联关系，比如所购商品；对典型行为进行用户画像，比如信用卡使用习惯或客户等待时间；预测数据项之间的链路，比如人与人之间的潜在社会关系；约简数据，使其更加容易管理或暴露隐藏信息，比如潜在电影偏好；在模型包含不同专业知识的前提下对模型进行组合，比如改善电影推荐的效果；从数据中提取因果结论，比如判断客户购买相同产品（在多大程度上）是因为其认识的人对他们的影响（病毒式营销的必要条件），还是因为熟人的品味相似（社会学中的常见现象）。扎实掌握这些基本原则能帮你理解更复杂的技术或技术组合。

注 11：虽然结束因果关系解读的条件已经超出了本书范围，但是如果有人给你看一个附有方程参数的因果关系解释回归方程，那么你可以询问这些系数的含义究竟是什么、它们又为何可以对方程进行因果解释，直到答案让你满意为止。对这样的分析而言，让决策者理解是放在第一位的，因此你必须把这样的结果弄明白。

第 13 章

数据科学和经营战略

基本概念：奠定数据驱动商业成功的原则；通过数据科学取得和维持竞争优势；对数据科学能力进行精心管理的重要性

本章将讨论数据科学和经营战略的相互作用，包括如何用数据科学选择所要解决的问题的高层视角。可以看出，数据科学的基本概念让我们能够清晰地考虑战略问题。同时我们也可以证明，这些概念总的来说有助于考虑商业上的策略问题，如评估来自顾问或内部数据科学团队的数据科学项目提案。本章还将详细探讨数据科学能力的管理。

我们越来越多地看到关于基于数据科学解决商业问题的新闻报道。如第 1 章所述，一系列因素使得与之前相比，当代企业所掌握的数据体量惊人。但仅仅拥有数据，并不能保证数据驱动决策的成功。企业如何最大程度地利用数据财富？答案自然多种多样，但重要因素有两个：首先，企业管理层必须具有数据分析思维；其次，企业管理层必须创造出一种有利于数据科学和数据科学家健康发展的文化。

13.1　数据分析式思维，终极版

虽然第一条准则并不意味着管理层必须是数据科学家，但要求他们必须充分理解基本原则，从而预见和 / 或领会数据科学所带来的机遇，为数据科学团队提供合适的资源，并乐于在数据和实验方面投入。此外，除非企业管理层中有一位经验丰富且注重实干的数据科学家，否则管理层必须仔细地引导数据科学团队，才能保证团队不会偏离通往有效的最终商业解决方案的路线。如果管理者不理解这些原则的话，这一点将非常困难。管理者还应具备向数据科学家提出探索性问题的能力，因为后者往往会困在技术性细节中。我们必须承认，团队中的每个角色各有长处和短处，而由于数据科学项目涉及一家公司的众多方面，因而一个多元化团队是必不可少的。正如我们不能指望管理者一定有很深的数据科学专业知识

一样，我们也不能指望数据科学家**一定**具备很深的商业解决方案专业知识。虽然如此，但一个高效的数据科学团队一定是双方的合作的产物，而且任意一方都必须对对方领域的基本内容有所了解。管理一个完全不懂基本商业概念的数据科学团队纯属徒劳，同样，数据科学家在完全不懂数据科学基本原则的管理层手下干活也会非常痛苦，甚至毫无成效。

举个并不罕见的例子，数据科学家经常遇到一些仅能（有时很模糊地）看到预测建模的潜在效益，却不能意识到要对合适的训练数据或评估程序进行投资的管理人员。在他们手下干活是件苦差。这种企业或许能“成功”部署一个准确的预测模型，且能形成可行的产品或服务，但注定无法超过愿意通过投资提升数据科学水平的竞争对手。

夯实数据科学基础有着更深远的战略内涵。虽然没有进行系统的科学研究，但丰富的经验告诉我们，随着执行者、管理者和投资者越来越多地接触数据科学项目，他们会看到越来越多的机会。这样的极端例子包括谷歌和亚马逊（网页搜索、亚马逊的产品推荐和其他服务背后都有着大量的数据科学内容）。两家公司最终都开发了后续产品，为其他企业提供了与大数据和数据科学相关的服务。许多（也许是大多数）面向数据科学的创业企业都使用亚马逊的云存储和云处理服务来完成某些任务。谷歌的“预测 API”的复杂度和实用性也在不断提高（尽管我们不清楚它的普及程度）。

虽然以上两个是极端案例，但其基本模式几乎存在于每家拥有大量数据的企业中。一旦培养出针对一种应用场景的数据科学能力，该能力在整个业务中的其他应用场景也就明晰了。Louis Pasteur 曾说过一句著名的话：“机会只眷顾有准备之人。”现代创造性思维所关注的是新的思维方式和对特定问题“饱和”式思索这两者的并置。通过（以理论或实操的方式）研究应用数据科学的案例，我们可以做好准备去迎接受益于数据科学的新问题的连接和机遇。

举个例子，20 世纪 80 年代末 90 年代初，一家最大的电信公司用本书中描述的技术，将预测建模应用到降低维修电话网络的费用和语音辨识系统的设计中。随着对用数据科学解决商业问题的能力的理解加深，该公司陆续将相似理念应用到决策中，比如将大量资本投资用于改善网络，以及减少其新兴无线业务中的欺诈现象。情况仍在继续发展，用于降低欺诈行为的数据科学项目发现，如果在欺诈预测模型中加入社交网络联系（依据相互通话数据）的特征，就能大幅提高模型反欺诈的能力。21 世纪初，该电信公司率先发明了用社交联系提高营销效率的方法——这种新方法与基于社会人口、地理和先期购买数据的传统定向市场营销相比，效果显著改善。然后，电信业开始把这些社交特征纳入流失预测模型，其结果同样令人满意。这样的思路扩散到了线上广告业，（在 Facebook，以及其他线上广告生态系统中的企业中）掀起了一阵基于线上社会关系数据的线上广告发展的热潮。

这轮发展的驱动力既来自于经验丰富且能着眼于商业问题的数据科学家，也来自于精通数据科学的管理者和企业家，因为他们在学术界和商业界的文献中看到了数据科学发展所带来的机会。

13.2 用数据科学取得竞争优势

企业越来越多地开始考虑是否能从数据和 / 或数据科学能力中取得竞争优势。因为这是一种重要的战略思维，不能浅尝辄止，所以我们将花些时间深入探讨。

数据和数据科学能力是（互补的）战略资产，而企业在何种情况下能用这样的资产取得竞争优势呢？首先，这项资产必须对企业有价值。这一条似乎显而易见，但请注意，资产对企业是否有价值，取决于该企业做的其他战略决策。跳出数据科学的语境，在 20 世纪 90 年代的个人计算机行业，戴尔在与行业龙头 Compaq 的竞争中取得了极大的优势，这要归功于戴尔使用了基于网络的系统，让客户能根据个人需求和喜好配置计算机。而 Compaq 却无法从该系统中获取同样的价值。一个主要原因是戴尔和 Compaq 实施了不同的战略：戴尔是一家直接向客户交付的计算机零售商，通过目录进行销售，而基于网络的系统在这样的战略下非常有价值；Compaq 则主要通过零售店销售计算机，因此基于网络的系统无法发挥其作用。而当 Compaq 试着复制戴尔基于网络的战略时，却受到了零售商们的强烈抵制。结论是，新资产（基于网络的系统）的价值取决于公司的其他战略决策。

这个示例说明，在业务理解环节，我们应仔细考虑数据和数据科学如何在商业战略背景下产生价值，以及其在竞争对手的战略背景下是否效果不变。使用这种方法，我们就可以识别潜在的机会和威胁。数据科学界中，与戴尔和 Compaq 的示例类似的是亚马逊和 Borders 的竞争。亚马逊很早就能根据用户的图书购买数据，向线上购物用户进行个性化推荐。虽然 Borders 也能够利用其用户的图书购买数据，但实体零售战略却让其无法同样顺畅地提供基于数据科学的推荐信息。

因此，竞争优势的先决条件是，资产必须在我们的战略条件下有价值。而第二条准则是：为获取竞争优势，我们的竞争者要么不能拥有某种资产，要么没有从该资产中获取同样价值的能力。我们应同时考虑（多种）数据资产和数据科学能力。我们的数据资产是否独一无二？如果不是，那我们是否有能比竞争对手更匹配资产的战略呢？或者我们是否能凭借更好的数据科学能力，比对手更好地利用数据资产？

考虑如何通过数据和数据科学取得竞争优势，反过来也是考虑我们是否在此方面处于竞争劣势。针对上一个问题，可能竞争者的答案是肯定的，而我们的却不是。下文将假设我们在寻求取得竞争优势，不过下文观点在假设相反的时候，即我们希望与某善用数据的竞争对手平起平坐时，也对称地适用。

13.3　用数据科学保持竞争优势

下一个问题是：如果取得了竞争优势，我们能否继续保持它？如果对手能轻而易举地复制我们的资产和能力，那么我们的优势将很快消失。这个问题尤为关键：如果竞争对手有比我们更丰富的资源，那么他们只要采取我们的战略，就能很快超过我们。

一个基于数据科学的竞争战略是，在竞争中始终领先一步，持续地投资新的数据资产、开发新技术和能力。虽然这种战略也许能让我们的业务令人兴奋地迅速增长，但一般很少有企业能做到这一点。比如，你必须笃定自己手下的数据科学团队是最优秀的，因为数据科学家的效率也存在较大差异，最好的会比一般水平的更有天赋。如果你的团队很优秀，你就会愿意相信自己处于领先地位。我们将在下文中更详细地探讨数据科学团队。

另一种在竞争中领先的方式，是通过使竞争者无法复制己方的数据资产或数据科学能力（或复制成本高昂）而保持竞争优势。通过这种方法保持优势的途径有很多。

13.3.1 令人敬畏的历史优势

历史环境可能会使我们的公司处于优势地位，而且竞争者取得同样地位的成本会非常昂贵。这一次，我们依旧可以把亚马逊作为一个出色的范例。在 20 世纪 90 年代的“互联网热潮”中，亚马逊以低于成本的价格售书，而投资者持续回报公司。这使得亚马逊累积了体量庞大的数据资产（比如用户购买偏好和线上产品评论的海量数据），从而开发出基于数据的有价值的产品（比如推荐和产品评分）。这些历史情境已经不再，即使竞争对手想通过连年低价售书来复制亚马逊的数据资产，投资者也不可能为他们提供同样水平的支持了（更不要说亚马逊如今已经不仅仅出售图书了）。

本例同时也说明，数据产品本身就可以提高竞争者复制数据资产的成本。消费者们重视亚马逊提供的数据驱动推荐和产品评价 / 评分，这就产生了转换成本。竞争者若想吸引亚马逊的顾客来自己店里购物，就必须向他们提供额外价值，即要么降低价格，要么提供亚马逊所不能提供的其他有价值的产品或服务。因此，当数据采集直接与数据产生的价值挂钩时，其所产生的良性循环就会让竞争者进入一个进退维谷的局面：他们既需要客户来获取必要的数据，又需要数据来提供等价服务，吸引客户。

企业家和投资者也可以换个角度思考这个战略问题。什么样的历史情境虽然现在存在但不会永远持续？什么又能让我们取得或构建比未来可能成本更低的数据资产呢？或者，什么能使我现在打造一个未来会昂贵得多（或不可能成功）的数据科学团队呢？

13.3.2 独一无二的知识产权

我们的企业可能拥有独一无二的智力成果。数据科学中的智力成果包括挖掘数据或使用模型的新技术，它们既可以是取得专利的，也可以是商业机密。在前一种情况下，竞争者要么无法（合法地）复制解决方案，要么因成本过高而难以复制解决方案，比如，他们需要取得我们技术的授权，或开发新的技术来绕过专利。而如果我们的智力成果是商业机密，那么竞争者可能不知道我们如何应用解决方案。在数据科学解决方案方面，其实际的机制往往是隐藏的，只有结果可见。

13.3.3 独一无二的无形抵押资产

竞争对手可能无法得知如何应用解决方案。对成功的数据科学解决方案而言，模型性能高（比如预测模型的高准确率）的实际原因可能是不清晰的。预测模型的有效性可能主要依靠问题设计、所创建的属性、多种模型的组合等。在实践中，竞争者往往不清楚如何达到这样的效果。就算公开算法的所有细节，要使实验室方案在实际生产中起作用，其关键也可能在于应用中的细节。

此外，竞争优势的基础也可能是无形资产，比如那种特别适合部署数据科学解决方案的公司文化。举个例子，欢迎业务实验和（严格）支持数据需求的公司文化，自然会形成数据科学解决方案容易成功的环境。或者，如果开发者有学习数据科学的动力，就不太可能会在工程上搞砸高质量的解决方案。还记得那条格言吗？“你的模型不是数据科学家设计的那个，而是数据工程师搭建的那个。”

13.3.4　优秀的数据科学家

我们的数据科学家可能比竞争对手的更好。数据科学家的质量和能力存在较大差别，即使在接受过最上乘训练的数据科学家中，也会有人同时具备天生的创造力、敏锐的分析能力、商业意识和耐心，并因此能比其他人提出更好的解决方案，这是数据科学圈子普遍认可的事实。

每年举办的 KDD Cup 数据挖掘大赛的结果就能说明能力上最极端的差异。每年，数据科学家的顶尖的职业团体 ACM SIGKDD 都会举办一次会议（ACM SIGKDD 知识发现与数据挖掘国际会议），而每年的会议都会举办一场数据挖掘比赛。一些数据科学家喜欢参与比赛，而这样的比赛有很多，甚至已经变成了众包业务（见 Kaggle），第 12 章讨论过的 Netflix 大赛就是最知名的比赛之一。KDD Cup 是数据挖掘比赛的鼻祖，自 1997 年开始每年举办一次。为什么该比赛和本节内容相关呢？因为世界上一些最优秀的数据科学家也会参加这些比赛，成百上千（依年份和赛题而定）的选手会尽力解决问题。如果数据科学家的才能呈均匀分布状态，那么很难想象我们会看到同样的选手反复的获奖。但是事实就是这样的。有些人总是出现在获胜团队的成员名单内，有时候是一连多年获奖，有时候则是在同一年中的多个问题中获奖（大赛可能会包含多个赛题）。[1] 这里的重点是，即使是最拔尖的数据科学家，其能力也存在巨大差异，KDD Cup 大赛的“客观”结果也显示了这点。结果是，凭借这种能力上的巨大差异，最好的数据科学家可以根据薪资、公司文化、发展机会等方面按照他们的意愿来选择就业机会。

对顶尖的数据科学家的大量需求，会强化这种数据科学家的能力差异。所有人都可以自称数据科学家，而很少有公司能真正评估可能被雇用的数据科学家是否符合要求，由此引发了另一个难题：公司至少需要有一位顶尖数据科学家，才能真正评估求职者的能力。因此，培养出强大的数据科学能力的企业，就会比聘请不到数据科学家的竞争对手拥有更显著、更持久的优势。而且，顶尖的数据科学家会互相吸引，这能进一步加强我们的优势。

我们还必须承认，数据科学在某种程度上是一种手艺。分析方面的专业知识需要花时间习得，单是读书或看视频课程并不能让人精通它们。这种手艺是通过经验学习的，最高效的学习方法类似经典贸易中的方法：想出人头地的数据科学家要去给大师当学徒。这些人可以追随重视应用的顶尖教授完成研究生课程，也可以在企业里和顶尖的业界数据科学家合作。等到学徒的本领足够精湛，成为“熟练工人”的时候，就能在团队中扮演更独立的角色，甚至独立领导项目了。很多高水平的数据科学家都愿意在职业生涯中以这种身份工作，其中一小部分人自身成为了大师，因为他们既能发现数据科学中的新机遇（稍后详细讲解）又精通理论和技术。这些人中的一部分又会雇用自己的学徒。理解这种学习路径有助于我们在招聘时集中精力，只去寻找曾经跟着顶尖大师学习过的数据科学家。你也可以将其灵活运用：雇一位顶尖的数据科学家大师，从而吸引其他高水平而有抱负的数据科学家来当他 / 她的学徒。

除此之外，顶尖的数据科学家还必须有强大的专业网络。此处的“网络”并不是指线上的专业网络系统，而是指，高效的数据科学家应该与数据科学圈子里的其他数据科学家有着

注 1：这并不是说 KDD Cup 获胜者必定是世界上最优秀的数据挖掘者，许多顶尖数据科学家从来没参与过这样的比赛，有的可能只参与过一次，然后就专心做别的事情了。

深厚的联系。这是因为数据科学的领域过于广大、内容过于丰富，个人无法全部精通，而顶尖的数据科学家却往往精通某些技术，并且熟知许多其他技术。（注意不要犯“百样通，无一精”的错误。）然而，我们并不希望精通某些领域的数据科学家强行用一种方法去解决所有问题。一位顶尖的数据科学家面对手头的问题会引入必要的专业知识。这一点很大程度上要借助于强大而深厚的专业人脉。数据科学家会互相求助来引导自己寻找正确的解决方案。而专业交际网络越强大，解决方案就越优秀。而且，最优秀的数据科学家往往也有最好的人脉。

13.3.5　优秀的数据科学管理

要想在业务中成功应用数据科学，更关键的因素可能是对数据科学团队的高水平管理。优秀的数据科学管理者尤其难以寻觅，他们需要充分理解数据科学的基础，甚至本身就是称职的数据科学家。他们还必须拥有普通人所难以拥有的一系列能力。

- 真正理解和领会业务需要，而且应该有能力预测业务需要，从而可以在与职责不同的同伴的相互交流中，产生有关数据科学新产品和新服务的思路。
- 与搞技术的和搞业务的都能顺畅沟通，且得到他们的尊重。这一点通常指的是能把数据科学术语（本书中尽量少涉及的内容）转化成业务术语，反之亦然。
- 协调在技术上很复杂的活动，比如根据业务限制或成本，进行多模型或多过程整合，要求数据科学管理者理解业务的技术结构，比如数据系统或生产软件系统，从而保证数据科学团队得出的解决方案在实际中确实有效。
- 能预见数据科学项目的结果。我们曾讨论过，相比其他商业活动，数据科学更像研发。一个数据科学项目是否能取得积极的结果，在一开始，甚至在项目进行中都是高度不确定的。尽管本书其他部分简要讨论了构造概念证明研究的重要性，然而这种研究的正面和负面结果都无法预测更大型项目的成功与否。它们只能指导对下一轮数据挖掘循环过程的投资（回忆第 2 章）。如果要从研发管理中寻求管理数据科学的方法，那么你会发现只有一个预测指标能可靠地预测一个研究项目的成功与否，而且这种预测**非常准确：**研究人员以前的成功经历。数据科学项目的情况也一样，有的人就是能凭直觉看出项目会不会成功。虽然没有对这种情形出现的原因进行仔细研究，但是经验告诉我们就是如此。正如在一些数据科学大赛中我们看到有人多次表现优异，我们也能看到有些人能多次预料到数据科学新机遇，并抓住它们取得成功。这是非常令人印象深刻的，因为有许多数据科学管理者连一个会成功的项目都没看出来过。
- 以上能力均需在公司文化之下培养。

最后，我们的数据科学能力对竞争对手来说可能难以复制，或复制成本过高，因为**我们可以雇用更优秀的数据科学家和数据科学管理者**。这可能要归功于我们对数据科学家极具吸引力的声誉和品牌——他们更喜欢在对数据科学和数据科学家友好的公司中工作。或者也可能是因为公司对数据科学家有某种更微妙的吸引力。因此，我们来继续探讨吸引高水平数据科学家的方法。

13.4 吸引和培养数据科学家及其团队

在本章开始时，我们说过，确保公司能够最大限度地利用数据资产的两个重要因素是：首先，企业管理层必须具有数据分析思维；其次，企业管理层必须培育出一种有利于数据科学和数据科学家繁荣发展的文化。如上文所述，高水平的数据科学家和普通水平的数据科学家的能力存在巨大差异，一个高水平的数据科学团队和单个高水平的数据科学家之间也有巨大差异。但如何保证我们对顶尖数据科学家有吸引力呢？又如何成立优秀团队呢？

这是一个在实践中很难回答的问题。在本书写作之时，顶尖数据科学家的缺口仍非常大，这导致他们的需求市场竞争非常激烈。最善于雇用数据科学家的企业是 IBM、微软、谷歌这一类的企业，他们用各种方式清楚地显示对数据科学的重视，包括工资、津贴和 / 或无形资产。无形资产包括一些无法忽略的因素，比如，数据科学家喜欢与其他顶尖的同行共事。某些人可能会说，他们是**需要**这么做，不是为了享受日复一日的工作。因为数据科学领域过于广博，多个数据科学家的集体智慧可以让他们在解决方案中应用更多种类的技术。

不过，即使在这样不利的市场里，也是有成功路径可寻的。比起在行业巨头里工作，许多数据科学家希望获得更多的个人影响力。其中有许多人希望能够承担更多责任（同时也相应获得更多的经验），在更广泛的范围内输出数据科学解决方案；有的希望担任一家企业的首席科学家，而且清楚成为小型的、更灵活的公司的首席科学家更顺理成章一些；有的希望成为企业家，而且清楚在创业公司中担任数据科学家的经历可以给他们带来无价的经验；有的则单纯地享受在快速增长的企业中工作的刺激感：在一家年增长 20% 或 50% 的公司中工作和在一家年增长 5% 或 10%（或完全不增长）的公司中工作，其体验是非常不同的。

鉴于以上所有情景，企业如果希望在雇用数据科学家方面具有优势，就必须创造出适合数据科学和数据科学家的企业环境。如果你的数据科学家团队人手不足，那就开动脑筋吧。鼓励你的数据科学家们参与当地的数据科学技术社群，甚至成为全球数据科学学术圈的一分子。



有关结果的发表

科学是一项社会性事业，优秀的数据科学家经常会通过发表其工作进展来参与社区活动。而企业有时候很难理解这一点，因为企业会觉得这是在自我损耗，或者是勾结竞争对手，透露企业机密。而另一方面，如果不让数据科学家这么做，企业就无法雇用或者留住优秀的数据科学家。其实发表成果对企业也有一定好处，比如增加宣传，扩大曝光，从外部验证内部创意，等等。这个问题虽然并没有很清晰的答案，但仍值得企业谨慎考虑。有的公司比较激进地为自己的数据科学创意申请专利，如果这些创意后来被证明的确具备创新性和重要性，那么学术发表就是理所应当的。

企业可以通过聘请学术数据科学家来支持企业的数据科学。这样做的方法有好几种。针对有兴趣在实际中应用其研究成果的学者，企业可以资助他们的研究项目。本书的两位作者在业界工作时，都曾资助过学术项目，这么做本质上扩展了他们的数据科学团

队，而且其成员会关注他们感兴趣的问题并不断互动。最好的安排（根据我们的经验）是将数据、资金和有趣的商业问题相结合。如果该项目最终成为了某顶尖院校博士生论文的一部分，那么对企业而言，其收益将远超成本。资助一名博士生的成本大约是5 万美元一年，而这只是聘请一位顶级全职数据科学家成本的一小部分。这里的关键在于企业要足够理解数据科学，选择合适的学者——其专业需与企业的问题相匹配。

另一种方法则非常划算：聘请一位或多位顶尖数据科学家作为科学顾问。如果这种关系的结构能使得顾问真正在问题解决方案上进行互动，那么那些没有资源或者影响力来聘请最优秀的数据科学家的公司，就可以大大提高其最终解决方案的质量。这些顾问既可以来自合伙企业，也可以来自与你的公司具有相同投资人或董事会成员的公司，还可以是有足够时间提供咨询的学者。

另一种完全不同的方法则是雇用第三方来处理有关数据科学的问题。第三方数据科学提供商种类繁多，既有专门从事商业分析的大型公司（如 IBM），也有专业数据科学咨询公司（如 Elder Research），还有仅帮助少数客户发展他们的数据科学能力的精品数据科学公司（如 Data Scientists，LLC），[2] 你可以在 KDnuggets 上找到大量数据科学服务公司以及各种其他数据科学资源。在寻找数据科学咨询公司时请注意，他们的利益与其客户的利益并不是始终一致的。尽管这一点对经验丰富的咨询服务用户来说显而易见，然而并非每个人都明白这一点。

精明的管理者会有策略地使用所有这些资源。一位首席科学家或一位得到授权的管理者，通常可以为一个项目组建一个比大多数公司所能雇用到的团队更强大、更多样化的团队。

13.5 检验数据科学案例分析

除了建立一个可靠的数据科学团队之外，管理者如何确保公司能够最好地应用数据科学？答案是要确保员工对数据科学的基本原理有一定的理解和认识。这样一来，整个公司的员工都会经常发现新的应用场景。

在掌握了数据科学的基本原则之后，确保自己取得成功的最佳方法是通过许多例子来学习数据科学在商业问题中的应用。阅读那些涉及实际数据挖掘过程的案例研究，并制定自己的案例研究。虽然实际操作数据挖掘是有帮助的，但更重要的是理解商业问题和可能的数据科学解决方案之间的联系。你处理过的不同问题越多，就越能一眼看出并充分利用机遇，进而利用数据中“存储”的知识和信息。通常同一个问题的定义仅需稍做改变，就可以通过类比应用于另一个问题。

要记住，本书中出现的示例是为了用于说明而选择或设计的。而现实中，业务和数据科学团队不但应该做好面对多种混乱情况和各种限制的准备，而且必须灵活地应对它们。有些时候，他们有大量的数据和数据科学技术可供使用。而其他时候，情况似乎更像是电影《阿波罗 13 号》中的关键场景。电影中，指挥舱发生故障，引发爆炸，导致宇航员被困在距地球约 40 万千米的太空。此时二氧化碳水平急剧上升，这可能导致他们无法活着返回地球。简而言之，因为手头物资有限，所以工程师需要想办法用一个大的立方体过滤器换

注 2：免责声明：作者与 Data Scientists，LLC 有关。

掉以前较小的圆柱体过滤器（就像把方形的榫头楔进圆形的卯眼里）。在关键场景中，总工程师把指挥舱里所有的“东西”都倒到了桌子上，然后告诉队员：“朋友们……我们得想办法只用**那个东西**，把**这个东西**塞进匹配**这个东西**的洞里。”实际的数据科学问题往往更像这种情况，而不是教科书里的情况。

比如，Perlich 等人（2013）描述了关于这种情况的研究。在为线上展示广告选择目标用户时，如果要获取足够的理想训练数据，成本就会奇高无比。然而，从各种其他分布中为了其他变量而获取数据成本要低得多。他们的解决方法非常有效，能把根据这些替代数据构造的模型拼凑起来，然后“转化”成目标问题中可以使用的模型。使用这些替代数据就能大幅降低获取数据的投资。

13.6　做好准备，接受来源各异的创意

一旦各方都理解了数据科学的基本概念，各种关于问题解决方案的创新概念就会开始从四面八方涌入。它们既可以来自发现了新业务线的执行官，也可以来自负责处理损益责任的主管，还可以来自负责管理业务流程的管理人员，以及详细了解具体业务运作流程的一线员工。我们应该鼓励数据科学家在整个业务过程中和员工保持交流，而且他们的绩效评估在某种程度上应该基于他们用数据科学创造的新思路是改进业务的效果。顺便提一下，这样做还会带来一些意料之外的收获：数据科学家拥有的数据处理技能通常可以被以不那么复杂的方式应用，这样可以帮助没有这些技能的其他员工。管理者通常不知道他们也可以获取一些数据——那些不需复杂的数据科学知识就能直接帮助管理者的数据。

13.7　做好准备，评估数据科学项目提案

通过数据科学改进商业决策的想法可以有很多方向。管理者、投资者和员工都应该有能力清晰地形成这样的想法，而决策者则应该做好准备来评估这些想法。本质上，我们应该既能制定切实的提案，又能评估这些提案。

第 2 章描述的数据挖掘过程提供了关于这一点的指导性框架。其过程中的每一步都会暴露一些问题，而这些问题不仅应该在制定项目提案时被考虑到，还应该在评估提案的环节被考虑到。

- 该商业问题是否界定明确？数据科学解决方案是否解决了该问题？
- 评估解决方案的方法是否清晰？
- 我们能否在对部署进行巨额投资之前看到成功的依据？
- 公司是否拥有它所需要的数据资产？比如，是否有供有监督建模使用的标注训练数据？如果没有，公司是否准备投资于数据资产？

附录 A 提供了一个起始问题列表，用于评估数据科学提案。这些问题是按照数据挖掘的流程整理的。请看一个说明性的例子。（附录 B 中提供了一个可供评估的提案示例，其内容有关用户流失问题。）

13.7.1 数据挖掘提案示例

你的公司开发的小部件 Whiz-bang® 目前有 90 万名安装用户，现在你要开发 2.0 版本，该版本的运营成本远低于现版本。你希望能把现版本的所有用户全部转移到（迁移到）2.0 版本。但由于 2.0 版本更换了界面，因而存在一个严重的风险：用户可能会因难以掌握新界面的使用方法而不愿换用新版本，进而对你的公司产生不满，甚至会因此转向竞争对手旗下的流行小部件 Boppo®。市场部设计了一种全新的迁移激励方案，其中每个目标用户的成本是 250 美元。但收到激励的用户是否一定会换用 2.0 版本，这无法保证。

一家第三方公司 Big Red 咨询公司给 Whiz-bang® 2.0 设计了一个选定目标用户的方案，而你因为展示出了出色的数据科学基础知识和数据科学能力，被选来评估 Big Red 的这项提案。Big Red 的选择看上去是否正确？

> **Whiz-bang 目标用户迁移方案——由 Big Red 咨询公司设计**
>
> 我们将用现代数据挖掘技术开发一个预测模型。上次会议中提到，我们估计用户迁移阶段的预算是 500 万美元。但若要调整预算，方案也可以很方便地随之调整。该预算下，我们可以选择 2 万名目标用户。以下是选择方法：
>
> 首先用数据构建模型，以判断用户是否会在受到激励后进行迁移。数据集包含用户的一系列属性，如前期与客服的互动次数和互动类型、小部件的使用程度、用户地址、对技术熟悉的程度的估计、作为公司用户的时间及其他忠诚度指标，如使用其他公司的产品或服务的数量。目标变量为用户在受到激励后是否会迁移到新版应用。我们将根据数据构造线性回归，以估计目标变量。我们将根据模型在该数据集上的预测精度，尤其是模型精度是否比随机选定目标用户的精度更高，来对模型进行评估。
>
> 模型的使用方法是：首先用回归模型估计每个用户的目标变量值，值大于 0.5，我们就认为该用户会迁移；否则就认为其不会迁移。然后，从被认为会迁移的用户中随机选择 2 万名，作为推荐的目标用户。

13.7.2 Big Red提案中的缺陷

我们可以运用对数据科学基本原则和其他基本概念的理解，找出提案中的缺陷。附录 A 提供了评估此类提案的初始指南，其中包括一些需要提问的主要问题。但总的来说，本书也可以被视作一本提案评估指南。以下是 Big Red 提案中的一些主要缺陷。

业务理解

- 目标变量定义不准确。比如，迁移必须在多长时间内发生？（见第 3 章）
- 数据挖掘问题的定义应该与商业问题更加匹配。比如，如果用户（或所有人）无论如何都会迁移呢（即使没有受到激励）？这种情况下，选择目标用户的激励成本就完全浪费了。（见第 2 章、第 11 章）

数据理解 / 数据准备

- 没有带标注的训练数据！由于这是一种全新的激励措施，因而我们应该花一些预算来获

取一些用户的标签数据。我们既可以通过（随机）选择一小部分用户作为激励目标来获取数据，也可以使用其他更复杂的方法。（见第 2 章、第 3 章、第 11 章）

- 如果担心会在无论如何都会迁移的用户身上浪费激励成本，那我们还可以在获取训练数据的一段时间内观察“控制组”的情况。这点应该不难，因为所有未被选为目标的用户都属于“控制组”。可以为“在不给出激励的条件下，用户是否会迁移”这一问题单独建模，并且根据期望值框架将这些模型进行组合。（见第 11 章）

建模过程

- 类别型目标变量不适合用线性回归建模，而适合使用分类方法，比如树型归纳、逻辑回归、*k*- 最近邻等。我们甚至可以尝试用多种方法建模，然后用实验方法对它们进行评估，以挑选出最佳方法。（见第 2 章、第 3 章、第 4 章、第 5 章、第 6 章、第 7 章、第 8 章）

评估环节

- 评估不应该使用训练数据，而应该使用一些保留方法（如，交叉验证 / 或前文探讨的阶段式方法）。（见第 5 章）
- 是否要对模型进行领域知识验证呢？如果验证后发现数据收集过程存在问题怎么办？（见第 7 章、第 11 章、第 14 章）

部署环节

- 在回归值大于 0.5 的用户中随机选择的方法并不明智。首先，无法确定回归值为 0.5 一定相当于迁移概率为 0.5；其次，0.5 这个值在许多情况下都很武断；最后，既然该模型会生成排序（比如根据迁移概率，如果我们使用更复杂的公式，那么还有可能根据期望值排序），我们就应该用这些排序来指导对目标用户的选择：在预算内选择排名最高的一些用户。（见第 2 章、第 3 章、第 7 章、第 8 章、第 11 章）

当然了，这只是提案中可能出现的缺陷的一个例子。提案不同，其缺陷也不同，发现缺陷所依据的概念也随之不同。

13.8　企业的数据科学成熟度

如果一个公司想切实地推行数据科学计划，那么坦率而理性地说，该公司需要评估自身的数据科学**成熟度**。虽然这一概念是一种自我评估指导，超出了本书的范围，但此处仍需简要介绍一下。

不同公司的数据科学能力在许多方面都存在巨大差异，比如一个对战略规划来说非常重要的方面：公司的“成熟度”。这个概念特指用于指导公司的数据科学项目的过程的系统性和有根据的程度。[3]

成熟度评估范围的一个极端是，公司的数据科学流程完全是随机的。许多公司中的员工在参与数据科学和数据分析项目时，并没有这方面的学习经历，而相关管理者也对数据科学的基本原则和数据分析式思维一窍不通。

注 3：对公司能力成熟度感兴趣的读者不妨在维基百科网站上阅读软件工程能力成熟度模型的相关内容“Capability Maturity Model”，这也许可以激发你对相关探讨的灵感。



有关“不成熟”公司的解释

“不成熟”并不表示公司必定会失败，而是指公司取得成功的变数很大。相较于成熟的公司而言，这样的公司更依赖运气来取得成功。其项目的成功也取决于那些碰巧数据分析思维天生敏锐的个人所付出的巨大努力。不成熟的公司可能会大规模地应用不那么复杂的数据科学解决方案，或者小规模地应用复杂的数据科学解决方案，却很少能大规模地应用复杂的数据科学解决方案。

中等成熟度的公司会雇用训练有素的数据科学家，也会雇用理解数据科学基本原则的业务经理与其他利益相关者。这两方都非常清楚如何用数据科学解决商业问题，且都会参与能直接解决商业问题的解决方案的设计与实施。

成熟度最高的公司会持续改进其数据科学**流程**（而不只是解决方案）。这些公司的高管不断挑战数据科学团队，以逐步向其灌输能够使其解决方案更好地与商业问题保持一致的流程。同时他们也会意识到，比起明年才能完成的理论上的最优方案，选择今天就能实现的次优方案才是务实的做法。公司内的数据科学家也会自信地认为，在提议公司通过投资改进数据科学过程时，他们的建议会得到开明的管理层的考虑。这并不是说每个这样的要求都会得到满足，但是该提案将根据其在业务背景中的优点受到评估。



数据科学既非运营亦非工程

把数据科学成熟度与软件工程的能力成熟度模型相类比其实不太恰当，因为这样的类比可能流于表面。那些适用于软件工程，甚至适用于制造或运营的流程，在数据科学领域并不奏效。而且，这样的做法可能会让优秀的数据科学家愤然离去，而管理者可能还不知道原因。解决问题的关键是要理解**数据科学流程**和做好数据科学工作的方法，并努力建立一致性，努力获取支持。记住，比起工程和制造，数据科学更像研发。举一个具体的例子，管理层必须及早并经常持续提供资源以对数据科学项目进行可靠评估。有时这涉及购买无法通过其他方法获取的数据。这还通常涉及分配工程资源来支持数据科学团队。反过来，数据科学团队应该尽力向管理层提供尽可能与实际商业问题相匹配的评估。

举一个具体例子，试想不同成熟度的公司会如何处理电信用户流失问题。

- 在不成熟的公司里，擅长分析的员工（有可能）会根据他们在客户流失管理方面的直觉，实施临时拼凑的解决方案。这样的方案可能奏效，也可能失败。不成熟的公司很难评估不同方法的效果，也无法判断自己的方案是否接近最优。
- 中等成熟的公司会在尽可能模拟实际商业环境的条件下，用定义明确的框架来测试不同的备选方案（比如，在试验平台上运行最近的生产数据，比较不同方法的效果），然后仔细考虑其中的成本和收益。

- 高度成熟的公司可能会用和中等成熟的公司完全相同的方法来判断最有可能离开的用户，甚至判断离开后会使公司蒙受最大期望损失的用户。他们还会努力实现整个流程，收集必要数据，判断激励的效果，从而找出受到激励后能带来最大期望价值提升（相比不受到激励）的用户。这样的公司可能还会把这样的程序融入用来评估不同激励策略或不同参数（如不同的折扣）的实验和 / 或优化框架。

虽然对数据科学成熟度进行坦率的自我评估并不简单，但是关键的是，要充分利用当前的能力，并进一步提高自己的能力。

第14章

总结

不能简明地解释一件事，说明你对它理解得不够。

——爱因斯坦

对数据科学实践最好的描述是分析工程和探索的结合。商业中会存在我们需要解决的问题，而该问题很少能直接与基础的数据挖掘任务相对应。我们通常会从现成的工具入手，把该问题分解为我们能够解决的子问题。至于那些我们不知道能解决到**什么程度**的问题，则需要通过数据挖掘和评估来观察。如果这一方法不奏效，那么可能需要用完全不同的方法继续尝试。整个过程中，我们既可能会发现有助于解决问题的知识，也可能会发现一些意想不到的东西，进而引导我们取得其他重要成功。

在考虑应用数据分析方法解决商业问题时，分析工程和探索缺一不可。缺少分析工程的结果是，数据挖掘的结论很可能无法用于解决商业问题。而如果没有将整个过程视作一个探索发现过程，常常会导致企业无法恰当地部署管理、激励和投资，进而导致整个项目失败。

14.1　数据科学的基本概念

理解和接受数据科学的基本概念，会使分析工程和探索发现更加系统化，更有可能取得成功。本书中介绍了一系列最重要的基本概念。我们将其中一部分概念直接作为章节标题，而其他概念则在讨论过程中自然而然地介绍到了（并不一定标记为基本概念）。从设想数据科学如何改进商业决策，到应用数据科学技术，再到部署结果以改进决策的过程，这些概念贯穿于整个过程之中。这些概念也可以支持许多商业分析。

基本概念大体可以分为以下三种。

(1) 关于如何将数据科学应用于企业和竞争格局的一般概念，包括如何吸引、构建和培养数据科学团队，如何利用数据科学带来竞争优势，如何保持竞争优势，以及做好数据科学项目的战术原则。

(2) 数据分析式思维的一般方法有助于我们收集合适的数据、构想合适的方法。这些概念包含**数据挖掘流程**、各种**高层次的数据科学任务**的集合，以及如下所述的原则。

- **在整个数据挖掘流程中，数据科学团队都应谨记亟待解决的问题和使用场景。**
- **数据应被视作资产，因此我们应谨慎考虑对其进行投资，以充分利用该资产。**
- **期望值框架有助于构造商业问题。它可以让我们看到商业问题中包含的数据挖掘问题，以及商业环境带来的成本、收益和约束。**
- **泛化能力和过拟合：如果过度仔细地观察数据，那么总能发现其中的模式。但是我们希望这些模式也能推广到新数据中。**
- **把数据科学应用到结构良好的问题中或探索性数据挖掘中时，需要在数据挖掘流程的不同环节付出不同的努力。**

(3) 从数据中实际获取知识的一般概念。这些概念也是大量数据科学技术的基础，包括以下几条：

- **识别富信息属性，即与我们关注的未知量相关或能提供其相关信息的属性；**
- **用数值函数模型拟合数据：选定目标（函数），并根据它选定一系列参数；**
- **对模型复杂度进行必要的控制，在泛化能力和过拟合之间找到平衡点；**
- **计算数据所描述的对象之间的相似度。**

我们发现数据科学的基本概念同样也是许多数据科学策略、任务、算法和流程的基础。本书反复强调，这些概念不仅能帮助我们进一步理解数据科学的理论和实践，还能帮助我们更加全面地理解数据科学的方法和技术，因为这些方法和技术往往就是一条或几条基本原则的特定实例。

我们知道，用期望值框架构造商业问题，有助于将问题分解成我们知道如何处理的数据科学任务，这一点在多种商业问题中都适用。

在从数据中获取知识时，我们发现，“判断两个由数据描述的对象之间的相似性”这一基本概念得到了直接应用，比如寻找与最佳客户最为相似的客户。这条概念既可以通过最近邻方法用于分类和回归，也是聚类（即对数据对象进行无监督分组）的基础。它还是根据查询语句寻找最相关文档的基础，也是不止一种常用的推荐方法的基础（比如，把用户和电影放在同一个“品味空间”中，然后寻找与某个用户最为相似的电影）。

说到度量，我们就想到**提升度**的概念。提升度被用来度量在多大程度上特定模式比随机情况更有可能出现。在数据科学中，这一概念在对多种模式进行评估时经常出现，比如通过计算目标群体中的提升度来评估精准广告的算法，比如判断支持或反对某结论的证据的权重，再比如判断某种重复出现的共现关系是有意义的，还是仅仅因为共现关系中的元素本身都很高频。

理解基本概念还有助于促进企业利益相关者和数据科学家之间的交流。这不仅是因为术语共通，更是因为双方对彼此的理解加深了。我们不会再错过讨论中的重要方面，而会深入挖掘并提出问题，以揭示原来极可能被忽略的重要方面。

举个例子，假设你的投资公司打算投资一家提供个性化网络新闻服务的数据科学公司。你想知道这种个性化新闻的实现方法，而对方声称自己使用了支持向量机。再假设本书中没有讲过支持向量机，而你的数据科学知识仍足以阻止你轻易认同对方的答案，而是胸有成竹地继续问："那是什么？"如果对方真的了解这项技术，就会根据基本原则做出一些解释（像第 4 章一样）。你现在可以接着问："你们要用什么训练数据？"这个问题不仅会给对方的数据科学家留下深刻印象，还能够判断对方的所作所为是可靠的，还是仅把"数据科学"当作障眼法。你可以继续思考："根据这些数据构造的预测模型（不管是什么模型）是否能解决他们的商业问题？"然后你可以继续问他们能否能找到此类问题所需的可靠的的训练数据，等等。

14.1.1　将基本概念应用于新问题：挖掘移动设备数据

我们反复强调过，只要把数据科学想成概念、原则和一般方法的集合，就能更加广泛地理解数据科学活动，并更加成功地将数据科学应用到新商业问题中。请考虑以下的新示例。

近期（在撰写本书时），消费者的线上活动开始显著地从传统计算机转移到种类更多的移动设备上。许多原本研究如何通过台式机触达用户的企业，如今开始争相学习通过移动设备触达用户，比如通过智能手机、平板电脑，以及随着 Wi-Fi 的普及而愈发常见的笔记本电脑。虽然我们不会讨论该问题中的复杂细节，但在我们看来，拥有数据分析思维的人应该能注意到，移动设备提供了一种新型数据，而这种数据的影响力目前仍未被充分开发。尤其是，移动设备会在其定位信息方面与数据产生联系。

比如，在移动广告生态系统中，根据个人隐私设置，移动设备可能会把我的实际 GPS 定位广播给想把我作为广告、每日特惠或其他促销活动的目标用户的企业。即使我不广播我的 GPS 定位，我的设备也会广播我现在使用的网络的 IP 地址，而这通常也会包含定位信息。

如何使用此类数据？让我们来应用基本概念。如果不想仅限于探索性数据分析，就必须根据具体的商业问题进行考虑。有些企业可能面临同样的问题，并且能够关注其中的一到两个。而企业家或投资者则可以借鉴多个其他企业或客户最近面临的各种问题。我们来选一个与这些数据相关的问题。

广告商在当今世界会面临这样的问题：移动设备多种多样，而一个特定用户的行为可能会分散记录在不同设备中。在台式机时代，只要广告商发现潜在用户，他们就能通过用户的浏览器 cookie 或设备 ID 采取相应的行动，比如展示精准广告。但在移动设备生态系统中，用户活动分散于多个设备，如果通过一个设备发现了潜在客户，那么如何通过他的其他设备对其展示精准广告呢？

一种方法是用定位数据将可能属于同一个用户的其他设备筛选出来。如果我们能刻画出某个移动设备的位置访问行为，就可以排除大部分可能的备选项。想必一个人的智能手机的定位信息应该会与其笔记本电脑的定位信息非常相似，在考虑到所使用的 Wi-Fi 地址时尤其如此。[1] 因此，我们可以利用评估数据项相似性的知识（见第 6 章）。

注 1：如果担心隐私泄露，那么可以对这些数据做匿名化处理。稍后将讨论更多细节。

在数据理解环节，我们需要决定如何确切地表示用户的设备及其定位。暂时放下算法和应用的细节，转而考虑基本概念，我们就会发现，尽管本例与文本完全无关，但文本挖掘示例（见第 10 章）的问题定义中出现的概念却非常适用于本例。在挖掘文本数据时，我们常常会忽略文本的结构，比如语序，而单纯把每篇文档视作来自一个可能很庞大的词汇表的单词的集合。这样的思路也可以应用于本例。显然，一个人访问的位置存在重要的结构，比如访问顺序，但对数据挖掘来说，最简单的策略往往就是最好的。类比第 10 章探讨的“词袋”表示法，我们先假设每个设备都是一个“定位袋”。

如果要找同一个用户的其他设备，不妨应用文本中的 TFIDF 概念。在专注于寻找同一用户的不同设备的相似性计算中，使用者众多的 Wi-Fi 地址（比如华盛顿广场公园转角处的星巴克）不太可能提供很多信息，因此这样的地址的 IDF 分值较低［可以把这里的“D”想成“Device”（设备），而不是“Document”（文档）］。另一种极端情况则是，由于许多人家中的 Wi-Fi 连接的设备数较少，因而更能区分同一用户的不同设备，而定位的 TFIDF 也能提升这些位置在相似性计算中的重要性。两种极端情况之间的一个例子是办公室 Wi-Fi 网络，其 IDF 值也会处于两者之间。

如果我们像第 10 章搜索爵士音乐家的示例一样，在 TFIDF 定义上应用相似性，用基于定位袋的 TFIDF 表示法作为设备的画像，那么就可以寻找与已识别为目标的设备最为相似的设备了。假设我的笔记本电脑就是已识别为目标的设备，它曾连接过我家的 Wi-Fi 网络和我办公室的 Wi-Fi 网络，同样连接过这两个网络的移动设备有我的手机、平板电脑，还有我妻子、几个朋友和同事的一些移动设备（但这些设备在其中一个 Wi-Fi 网络上的 TF 值会低于我的设备）。因此，我的手机、平板电脑很可能和我的笔记本电脑非常相似（可能最为相似）。如果广告商认为我的笔记本电脑很适合投放某支广告，那么根据以上推断，我的手机和平板电脑也同样如此。

本例的意图不在于明确地找到不同移动设备的对应用户[2]。本例展示了一个概念性工具包是如何帮助考虑一个全新问题的。一旦这些思维得到概念化，数据科学家将应用我们讨论过的许多概念（例如如何评估替代实施方案），深入研究真正有效的方法以及如何充实和扩展这些想法。

14.1.2　改变对商业问题解决方案的思考方式

本例同时也提供了对另一个重要基本概念的具体说明（即使经过这么多页的详细介绍，我们也讲不完它们）。一种普遍情况是，在数据挖掘流程中的业务理解 / 数据理解环节中，“问题是什么”的概念变成了“我们到底能对数据做什么”。这个转变往往很细微，但我们必须（努力）关注这个转变。为什么？因为所有利益相关者都没有参与数据科学问题的定义过程。如果我们忘记问题已经发生转变（尤其是转变非常细微时）再往下进行就会遇到阻力。而这种阻力可能仅仅是由于误解而产生的！更严重的是，我们可能会认为这种阻力是由于固执而产生的，因而引发不愉快，最终导致项目失败。

回头继续考虑选择目标移动设备的示例。敏锐的读者可能会说：“等等，我们一开始要找的是使用不同设备的同一用户。而我们通过设备的定位信息找到了非常相似的用户。我不

注 2：但这仍是最杰出的一家移动广告公司所实现的真实解决方案的精髓。

否认这些相似用户的集合中很可能包含同一用户（比我能想到的任何替代方案更可能）但这与在不同设备上查找同一用户是不同的。”这个读者没说错。在问题定义环节，问题出现了轻微的改变。现在我们把识别同一用户概率化了：虽然具有高度相似的位置画像的一些设备很有可能属于同一用户，但是我们不能完全确定这一点。我们必须清楚这一点，并且要跟利益相关者交代明白。

事实证明，在进行精准广告或促销时，这样的改变能被所有利益相关者接受。回想一下评估数据挖掘解决方案的成本 / 收益框架（见第 7 章），显然，对许多促销活动来说，把假阳性个体选为目标的成本，会比选中真阳性个体的收益低一些。而且，如果在促销中每次“误选”都能碰巧选到兴趣相同的其他用户的话，那么实际上许多促销方很乐于“误选”。我的妻子、好朋友和一些同事与我的品味和兴趣相似，是很好的促销目标！[3]

14.2　数据做不到的：圈中人回顾

本书关注的是通过加强数据驱动决策，我们如何、为何以及何时能从数据科学中获取商业价值。我们还需要考虑数据科学与数据驱动的决策的局限性。

有些事情是计算机擅长的，有些则是人类擅长的，而这两者往往不尽相同。比如，人类更擅长从全世界的所有东西中区分出一些相关联的方面，并从中收集数据来支持特定任务。而计算机则更擅长从包括大量（可能）相关变量的浩如烟海的数据中筛选重要信息，以及通过量化变量相关性来预测目标。



《纽约时报》社论版专栏作家 David Brooks 撰写过一篇优秀的文章，题为“What Data Can’t Do”（Brooks, 2013）。如果你打算用神奇的数据科学来解决问题，那么不妨读一读这篇文章。

数据科学是人类智慧和计算机技术的明智组合，能做到两者中任何一方不能单独做到的事情。（所以要当心那些夸下海口的工具供应商！）第 2 章介绍的数据挖掘流程有助于指导人类和计算机的这种组合，而该过程所引入的结构强调人类之间的早期交互，从而确保了数据科学方法围绕着正确的问题应用。检查数据挖掘流程也能说明，人际互动不仅在任务选择和问题定义环节起关键作用。如第 2 章所讨论的，人类的创造力、知识和常识发挥作用的一个环节是选择正确的数据进行挖掘，而这一环节（特别是考虑它的重要性时）在数据挖掘的讨论中经常被忽略。

人际互动也是评估环节的关键。合适的数据与数据科学技术的组合能出色地选出将客观标准最优化的模型。而只有人类能分辨对于特定问题而言，什么是最优化的最佳客观标准。这涉及大量人类的主观判断，因为通常真正的最优化标准是无法度量的。因此人类必须尽可能找出最好的替代标准，并且牢记这些决定，因为它们可能是模型部署时的风险来源。然后，我

注 3：Crandall 等人（2010）在《美国国家科学院院刊》上发表的文章表明，人与人之间的地理共现情况能在很大程度上表明两人是否是好友：“如果两个人会在几乎相同的时间出现在几个特定地点，那么这两人就有很高的条件概率在社交网络中存在直接关系。”这意味着，即使是因为地理相似性而出现的“误选”，也在社交网络定位中存在一些优势。这一点在营销中非常有效（Hill 等，2006）。

们需要仔细地、有时有创造力地关注最终生成的模型或模式是否真的能解决问题。

我们还要记住，要应用数据科学技术的数据是包含人类决策的某个过程的产物。我们要摒弃“数据代表客观真理”的想法。[4] 数据包含了设计数据采集系统的人的信念、目的、偏见和语言用法。而数据的含义则会受到我们自身信念的影响。

考虑以下简单示例：许多年前，本书的两位作者以数据科学家的身份在最大的电信公司之一共事，彼时的无线业务出现了严重的欺诈问题，我们把数据科学方法应用于包括手机使用、社会呼叫模式、访问地址等的海量数据（Fawcett & Provost, 1996, 1997）进行分析。检测欺诈行为的模型中一个看似表现良好的部分表明，“从 0 号基站打来的用户的欺诈风险显著增大”。这一点通过谨慎的保留验证得到了证明。所幸（在本例中），我们进行了良好的数据科学实践，在评估环节进行了模型的领域知识验证。我们很难理解模型的这一部分，因为尽管有许多基站都显示欺诈概率上升[5]，但 0 号基站的表现最为“突出”。而且，其他基站出现这种情况很合理，因为只要查一下它们的位置就会找到说得通的理由，比如该基站位于犯罪高发地区。而如果我们查询 0 号基站的相关信息，却会发现什么都查不到，它甚至不在基站清单中。于是我们找顶尖数据大师指点迷津——**0 号基站的确不存在**，但数据中的确存在许多从 0 号基站打来的欺诈电话！

长话短说，我们对数据的理解出错了。简而言之，在用户账户上的欺诈情况得到解决前，通常要经历打印账单、寄出账单、用户收到账单、打开、阅读、采取行动等一系列过程，而在这段时间内，欺诈活动仍在继续。而欺诈情况被检测出后，这些通话就不应再出现在该用户下个月的账单中，因此我们需要从计费系统中将它们删除。但这不意味着它们被丢弃了，相反，它们会被保存到另一个数据库中（对数据挖掘工作而言这很幸运）。但不幸的是，设计该数据库的人认为某些域的数据没必要保留，其中之一就是基站编号。因此，当我们为建立训练集和测试集而调用所有诈骗电话数据时，所得到的数据中包含了这些通话。但因为它们没有基站编号数据，所以另一个设计决策（有意或无意地）导致这些域被填上了 0。因此，许多诈骗电话看似都来自 0 号基站！

这就是第 2 章介绍的“漏洞”。你可能觉得它们很容易察觉，实则不然，原因如下。试想数千万用户在这么多个月内会打多少电话，而每通电话又包含多少可能的描述性属性。我们不可能手动检验数据。而且，因为电话按用户分组，所以来自 0 号基站的电话不会大量聚集在一起，而是穿插在每个用户的其他电话之中。最后一点，可能也是最重要的一点，在数据准备环节，为了提高目标变量的质量，我们进行了数据清洗，因为有些被标为“欺诈”的电话实际上并非真的是欺诈电话。其中大部分可以因发现用户在先前未出现欺诈的时间段打过电话而洗清嫌疑。结果是，虽然来自 0 号基站的电话欺诈概率上升，但这并非预测欺诈的完美方法（而是危险信号）。

展示这个小案例的目的在于说明：“数据是什么”只是我们所做的解释。这个解释通常会在数据挖掘流程中发生改变，而我们需要接受这种可塑性。这个欺诈检测示例展示了对数据项解释的改变。当发现数据采集过程中的偏差时，我们通常会改变对数据采样的理解。比

注 4：爱好哲学的读者不妨阅读 W. V. O. Quine（1951）的经典文章“Two Dogmas of Empiricism”，作者在文中尖锐批评了将经验和分析分割开来的理念。

注 5：技术上，需要有更多从这些基站播出的电话性质出现显著变化，模型才能发挥最大用处。如果你感兴趣，我们的论文对此进行了详细探讨。

如，如果想对用户行为建模，进而设计或投放营销活动，我们就必须准确理解所要取样的用户群。这一点在理论上很浅显，但实际中它涉及对产生数据的系统和业务进行深入分析。

最后，我们需要能够识别**可以因数据科学甚至是人类的参与而增值的问题**。你可能会问："我们真的有足够的与手头要做的决定相关的数据吗？"我们可能需要在这个独特背景下做出非常高层次的战略决策。数据分析和理论模拟能提供深层的见解，但若要做出最高层面的决策，决策者必须凭借自己的经验、知识和直觉。这当然也适用于战略决策，比如是否要收购某个公司。虽然数据分析能支持决策，但毕竟每个情况都是独特的，因而必须依赖经验老道的战略家来做出决策。

这种有关独特情境的想法应该贯彻到底。举一个极端的例子，请考虑乔布斯的一句名言："根据受众需要去设计产品其实是非常难的，因为很多情况下，人们并不知道自己想要的是什么，而是需要你展示给他们看……但这不代表我们不用理睬用户的意见，而是表示他们很难在从未见过类似事物的情况下描述出他们想要什么。"放眼未来，随着精细自动实验能力的提升，我们有望可以不再询问用户的喜好和建议，而是通过观察得出用户喜好和建议。为此，我们需要遵循基本原则：把数据视作需要投资的资产。第 1 章中的 Capital One 就是一个范例：创造出很多产品，并投资于数据与数据科学来判断用户想要哪些产品，以及每个产品适合哪些用户（即在哪些用户身上是有利可图的）。

14.3 隐私、道德和挖掘个人数据

挖掘数据，尤其是个人数据，会引发不容忽视的道德问题。虽然新闻界和政府部门最近对隐私和数据（尤其是线上数据）问题进行了大量讨论，但问题的范围比这要广泛得多。许多面向消费者的大企业会采集或购买用户的详细数据，并将其用于本书中所讨论的许多商业应用决策。我们是否会被授信？如果会，那么我们的信用额度是多少？我们会被当作营销目标吗？我们想在网页上看到什么样的内容？我们应该被推荐什么产品？我们是否可能转投对手公司？我们的账户上是否存在欺诈情况？

隐私和改善商业决策之间的关系非常密切，因为愈发频繁的个人数据使用和愈加高效的商业决策之间似乎存在直接关系。比如，多伦多大学和麻省理工大学的研究员进行的一项研究表明，在欧洲颁布严格的隐私保护法律后，线上广告明显不如以前有效了。具体来说，"被展示广告的客户和未被展示广告的客户之间的购买意向的差异下降了约 65%，而欧洲之外的任何国家都没有出现这种情况"（Goldfarb & Tucker, 2011）。[6] 该现象不仅仅出现在线上广告业。如果将个人的传统数据加上详细的社交网络数据（比如谁与谁取得联系），那么欺诈检测（Fawcett & Provost, 1997）和目标市场营销（Hill 等 , 2006）的效率就能大大提高。一般来说，收集到的个人数据越详细，相关商业决策的质量就会越高。越来越少的隐私和越来越高的企业业绩之间看似有直接的关系，这从隐私和业务两个视角（有时来自同一人）同时引发了强烈的情绪。

该问题的解决不但远远超出本书范围，而且极其复杂（比如："匿名化"要到什么程度才可以？）且多样化。合理进行隐私友好型数据科学设计的最大阻碍，可能是难以定义隐私。

注 6：参见 Mayer 和 Narayanan 的网站（http://donottrack.us/bib/#sec_economics）阅读对此的批评，以及其他研究者关于行为定向在线广告的价值的文章。

Daniel Solove 是关于隐私的世界权威，其文章“A Taxonomy of Privacy”（2006）的开头如下：

> 隐私是一个混乱的概念，我们无法明确其含义。正如一名评论员所发现的，隐私深受“含义尴尬”之害。

Solove 的文章接下来用 80 页的篇幅对隐私进行了分类。Helen Nissenbaum 是另一名隐私方面的世界权威。她最近特别关注隐私和大规模数据库（及其挖掘）的关系。关于这个主题，她写了一本书，*Privacy in Context*，超过 300 页（非常值得一读）。提到这两个人是为了强调，隐私问题既不易理解，也不易处理，甚至不是仅用数据科学教材的一节或一章就能详细说明的。如果你是数据科学家或数据科学项目中的企业利益相关者，那你就应该关注隐私问题，并且花大量时间仔细考虑它。

14.4 数据科学是否还有更多内容

虽然本书已经很厚了，但我们仍尽己所能地选取了最有助于数据科学家和企业利益相关者理解和交流数据科学的最相关的基本概念。当然，本书并没有包含数据科学的所有基本概念，有的数据科学家可能会怀疑我们是否选择了最恰当的概念。但必须承认的是，本书包含了一些支撑着数据科学的最重要的概念。

有许多高级主题和与之密切相关的主题是根据本书所提出的基本概念建立的。这里不会列出它们——如果你感兴趣，那么不妨仔细研读近期的顶级数据挖掘研讨会议中的项目，如 ACM SIGKDD 数据挖掘和知识发现国际会议，或 IEEE 国际数据挖掘会议。这两个会议包含顶级行业跟踪，关注数据科学在商业和政府问题中的应用。

关于在进一步探索时可能会发现的那类话题，我们再举一个具体的例子。回忆数据科学的第一条原则：数据（和数据科学能力）应被视为资产，且应被作为待选的投资对象。本书逐层深入探讨了投资数据这一概念。如果我们在数据科学项目中明确地应用“考虑成本效益”这一一般框架，就能产生新的思路。

14.5 最后一例：从众包到云包

互联网带来的企业和“消费者”的互通，改变了劳工经济。基于网络的系统，如亚马逊的 Mechanical Turk 和 oDesk 等，促进了一种可以被称为“云劳动”的众包业务——通过互联网来控制大量独立承包商。一种与数据科学紧密相关的云劳动是“微外包”：将大量小而定义明确的任务外包。微外包与数据科学紧密相关，因为它改变了数据投资的经济条件和可行性。[7]

例如，回忆有监督建模的应用条件（见第 2 章）。我们不但需要准确定义目标变量，而且需要知道训练数据的目标变量值（“标签值”）。有时我们可以做到前者，但手头却没有标签数据，此时就可以用微外包系统（比如 Mechanical Turk）来标注数据。

比如，广告商想避免在令人反感的网页（比如那些包含仇恨言论的网页）上投放广告，但

注 7：感兴趣的读者可以访问 Google Scholar，搜索“data mining mechanical turk”，或含义更广泛的“human computation”，寻找相关论文，并可以点击前向引用链接（Cited by）获取更多信息。

他们如何在数以亿计的备选网页中找出这些网页呢？让员工一一检查的话，成本太高了。你可能立刻会想到，可以用文本分类方法（见第 10 章），获取网页的文本，并如先前所述，用特征向量对其进行表示，然后构造一个仇恨言论分类器。但很可惜，我们没有仇恨言论网页的代表性样本，因而没有训练数据。但如果这个问题足够重要[8]，我们就应考虑投资于标注训练数据，看看能否构建一个能判断网页是否包含仇恨言论的模型。

在获取标注训练数据的示例中，云劳动改变了对数据进行投资的经济条件。我们可以通过互联网雇用廉价劳动力，以多种方法获取数据。比如，我们可以让亚马逊 Mechanical Turk 的员工给网页加上令人反感与否的标签，来给我们提供目标标签，这种方法比雇用学生便宜得多。

> 每名经过训练的实习生完成该工作的速度是每小时 250 个网页，成本为每小时 15 美元。而将该任务发布在亚马逊 Mechanical Turk 时，打标签的速度提升到了每小时 2500 个网页，而总成本却不变。（Ipeirotis 等，2010）

问题是，一分钱一分货，低价有时代表低质量。近 5 年来，出现了大量关于如何在利用云劳动的的同时保持质量不变的研究。注意，给网页加标签只是用云劳动增强数据科学的一个例子。即使在这个案例研究中也存在许多其他选择，比如用云劳动查找仇恨言论的正样本个体（Attenberg & Provost, 2010），而不是给我们所提供的网页加标签。云劳动也可以用于在游戏式的系统中找到当前模型出错的地方，即“打败机器”（Attenberg 等，2011）。

14.6　最后的话

20 多年来，本书的两位作者一直致力于将数据科学应用到实际商业问题中，这几乎已经成为了他们的第二天性。对我们来说，掌握这些明确的基本概念也非常有用。每次你在思考过程中陷入僵局时，只要想想这些基本概念，就能拨云见日。像“嗯，先回顾一下业务理解和数据理解吧……我们到底要解决什么问题”这样的过程就能解决许多问题，比如：是否决定研究期望值框架，是否要仔细考虑数据采集方式，是否明确定义了成本效益，是否要进一步投资数据，或是否恰当地定义了该问题的目标变量，等等。了解不同的数据科学任务，可以防止数据科学家用其所掌握的一种方式来应对所有商业问题。在考虑评估和用于比较的“基线”时，仔细考虑商业问题中的重要因素，能大大促进数据科学家与利益相关者的交流。[将直接汇报对商业问题无意义的统计量（比如均方误差）与这种方法对比，你就能看出差异。] 数据分析思维不仅能帮助数据科学家，也能帮助所有参与该过程的人。

如果你不是数据科学家，而是一位企业利益相关者的话，那么千万别让那些所谓的“数据科学家”用术语把你搞得云里雾里，本书中的概念加上你自己的商业知识和数据系统知识，能让你理解 80% 甚至更多的数据科学内容，进而提高你的工作效率。在读过本书后，如果你还不明白某个数据科学家说的话，那就要当心了。虽然数据科学中的确存在大量复杂概念，但优秀的数据科学家应该能够用本书中的术语来描述问题和解决方案的基本原理。

如果你是数据科学家，那么请接受我们的挑战。仔细考虑为什么你的工作对业务有帮助，并且将其展现出来。

注 8：事实上，广告出现在令人反感的网页上这个问题价值 20 亿美元（Winterberry Group, 2010）。

附录A
提案评估指南

高效的数据分析思维有助于系统地评估潜在的数据挖掘项目。本书中的材料应该已经给你提供了用于评估数据挖掘提案和发现其中的潜在不足的必要背景，这项本领既可以用于对自己的提案进行自我评估，也可以用于评估公司内部数据科学团队或外聘顾问的提案。

下文中包含了一系列在考虑数据挖掘项目时应想到的问题。这些问题根据第 2 章详细介绍的数据挖掘流程设计，是贯穿本书的概念性框架。在读过本书后，你应该有能力在概念上把这些问题应用到新的商业问题中。虽然下文的列表并非面面俱到（本书本来也没打算做到面面俱到），但仍包含了一些最重要的问题。

本书自始至终关注的都是数据科学项目，其重点是从数据中挖掘出规律、模式或模型，而这篇提案评估指南就反映了这一点。在有的数据科学项目的组织中，规律可能并不明显，比如，许多可视化项目一开始并没有清楚地定义建模目标。然而，数据挖掘流程能让我们将针对此类项目的数据分析思维结构化——这些项目比起有监督数据挖掘，更像无监督数据挖掘。

A.1　业务和数据理解

- 需要解决什么商业问题？
- 数据科学解决方案是否适合解决本问题？**注意：有时我们必须审慎地取近似。**
- 某个实例 / 示例对应什么业务实体？
- 问题是有监督的还是无监督的？
  - 如果是有监督的，那么：
    - 是否有**目标变量**？
    - 如果有，是否定义明确？

- ♦ 思考其如何取值。

- 属性是否定义明确？
  - 思考其如何取值。
- 针对有监督分类问题，对目标变量建模是否能实际改善本商业问题？或能否改善某个重要的子问题？如果是后者，那么其他子问题是否也能得到解决？
- 用期望值定义问题是否有助于将要解决的子问题结构化？
- 如果问题是无监督的，那么是否存在定义明确的“探索性数据分析”路径？（也就是说，分析的方向是什么？）

A.2 数据准备

- 获取变量值、构造特征向量并将其编入表格的做法是否切实可行？
- 如果不可行，那么是否存在其他定义清晰明确的数据格式？该格式是否考虑了项目的后期阶段？（许多后期的方法 / 技术都假设数据集是特征向量形式。）
- 如果建模过程是有监督的，那么目标变量是否定义明确？获取（训练集和测试集的）目标变量值并制表的方法是否明确？
- 如何获取目标变量值？该过程是否存在成本？如果是，那么提案中是否包含了这些成本？
- 从总体中获取的数据是否与将应用模型的数据相似？如果存在差异，那么提案中是否注明了选择性偏差？是否存在弥补该偏差的方案？

A.3 建模

- 选择的模型是否适用于目标变量？
  - 分类、类概率估计、排序、回归、聚类？
- 模型 / 建模技术是否满足任务的其他要求？
  - 泛化能力、理解能力、学习速度、应用速度、要求的数据量、数据类型、缺失值？
  - 该建模技术是否与问题的先验知识相符（比如，明明是非线性问题却要应用线性模型）？
- 是否应该尝试多个模型并进行比较（在评估阶段）？
- 针对聚类方法：是否定义了相似性测度？该测度是否对本商业问题有意义？

A.4 评估和部署

- 是否有进行领域知识验证的计划？
  - 领域专家或利益相关者是否会在模型部署前检验模型？如果是，那么模型的形式是否易于他们理解？
- 评价机制和指标是否适用于该业务问题？请回忆问题的初始规范。
  - 是否将经营成本和收益考虑在内？
  - 针对分类方法，如何选择分类阈值？
  - 是否直接应用了概率估计？

- 排序是否更恰当（比如，对固定预算而言）？
  - 针对回归方法，如何评估数值型预测的质量？为什么该方法适用于本问题？
- 评估阶段是否使用了保留数据集？
  - 可以使用交叉验证。
- 比较结果使用的基线是什么？
  - 在本问题背景下，该基线为何有意义？
  - 是否存在客观评估基线法的方案？
- 针对聚类方法，如何理解聚类结果？
- 根据计划部署方案是否能（最好地）解决本商业问题？
- 如果需要向利益相关者申请项目经费，那么度量最终（部署的）业务影响的方案是？

附录B

另一个提案示例

附录 A 提供了用于评估数据科学提案的一系列准则和问题，第 13 章中展示了一个用户迁移活动的提案示例（见 13.7.1 节），并指出了该提案的不足（见 13.7.2 节）。

本书中通篇使用电信公司用户流失问题的示例，本章将展示基于该问题的第二个提案示例及其评估。

情景和提案

你在 Green Giant 咨询公司（GGC）得到了一份好工作，管理一个刚刚学会数据科学技能的团队。GGC 正准备向 TelCo 发送一份提案，解决这家全国第二大的无线通信公司的用户流失问题。你团队里的分析师撰写了以下提案，你在把该提案呈给 TelCo 之前需要审核一遍。该提案是否有不足？你有什么改进建议吗？

通过有针对性的激励来降低用户流失——GGC 提案

> 我们认为，TelCo 应通过用户流失预测分析，测试其控制用户流失的能力。其核心思想是先利用用户行为数据来预测用户何时会离开公司，然后对这些用户有针对性地提供特殊激励，从而将他们留在公司。我们建议使用以下建模方法，该方法用 TelCo 现有的数据就可以实现。
>
> 考虑到保留合约到期后长时间逐月续订服务的用户的问题，我们将模拟用户在合约到期后 90 天内离开（或留在）公司的概率。我们认为，90 天的窗口期是预测用户流失的恰当起点，从中得到的经验也可以应用于其他流失预测问题。本模型将根据过去离开公司的用户的数据构造，而流失概率则会根据合约到期前 45 天内的数据进行预测，以便给 TelCo 留出足够时间向用户提供优惠激励。我们将通过构建集成树型模型（随机森林模型）来模拟流失概率，该方法以精度高、适用范围广而知名。

我们预计能辨别出 70% 在 90 天的窗口期内将离开公司的用户。这一点将通过在数据库上运行模型来验证。在与 TelCo 的利益相关者交流的过程中，我们了解到，所有用户维持的新程序都需要用户维持副总裁签字通过。而她指出，她将根据自己对本程序是否有意义的判断和公司中一些用户维持专家对本程序的意见做出决定。因此，我们将给副总裁和专家查看用户维持模型的权限，从而使他们能够评估本模型是否适用以及是否高效。我们提议，每周运行一次本模型，以估计合约将在 45 天内（上下浮动一周）到期的用户的流失概率。我们将按流失概率对用户进行排序，前 *N* 名将被作为目前激励的目标，*N* 的具体大小要视每周用户维持预算而定。

GGC提案的不足

我们可以用对数据科学基本原则及其他基本概念的理解，找出提案中的不足。附录 A 提供了一份评估此类提案的起步“指南”，其中包含许多主要问题。而且，本书本身就可以视作一份提案评估指南。下面是 Green Giant 提案中一些最严重的不足。

(1) 该提案目前只提到了根据“已离开公司的客户”进行建模。在训练（和测试）模型时，我们还需要获取**并未**离开公司的客户数据，以便模型找到用于区分两者的信息。（见第 2 章、第 3 章、第 4 章、第 7 章）

(2) 为什么要按流失概率从高到低排序，而不是在计算标准期望值后，按期望损失排序？（见第 7 章、第 11 章）

(3) 对最有可能受激励的（积极）影响的用户建模，不是更好吗？（见第 11 章、第 12 章）

(4) 如果要按第 3 条的思路往下走，那么我们可能没有所需的训练数据，需要通过购买来获取数据。（见第 3 章、第 11 章）

注意，目前的提案很可能只是完成业务目标的第一步，但我们必须讲清楚，**要注意观察我们能否准确估计流失概率**。如果能，就可以继续进行；如果不能，就需要重新考虑是否要对该项目进行投资。

(5) 提案中并未提到评估模型的**泛化能力**（即进行保留评估）。他们似乎要用训练集进行测试（“……在数据库上运行模型……”）。（见第 5 章）

(6) 提案中并未定义（提都没提）所要使用的属性！这仅仅是一个疏忽吗？还是因为该团队没有考虑到这一点？他们的计划是什么？（见第 2 章、第 3 章）

(7) 该团队如何是估计出模型 70% 的精确度的？提案中并未提及他们进行了初步研究，也没有对数据样本绘制学习曲线，也没有其他任何论据。因此该声明感觉像是猜的。（见第 2 章、第 5 章、第 7 章）

(8) 而且，在不讨论误差率或假阳性、假阴性概念的前提下，“辨别出 70% 将离开公司的用户”的含义并不明确。如果只字不提假阳性率，那么我完全可以说每个用户都会离开公司，从而使辨别率高达 100%。因此只有在提及假阳性率的情况下，谈论真阳性率才有意义。（见第 7 章、第 8 章）

(9) 为什么只选择一个模型？我们可以用现代工具包来方便地比较多个模型在同一组数据上的效果。（见第 4 章、第 7 章、第 8 章）

(10) 用户维持副总裁必须签字通过本程序，并且她指出她将亲自检验本程序是否有意义（领域知识验证）。然而，组合树模型对她来说是黑箱模型，提案中完全没提到该如何让她理解该过程辅助决策的原理。根据她的需求，我们需要牺牲一些精度，构建一个更易于理解的模型。一旦她“上了道”，我们就能用易理解性较差但精度更高的模型了。（见第 3 章、第 7 章、第 12 章）

术语表

注：本术语表是对 Ron Kohavi 和 Foster Provost（1998）编纂的术语库的扩展，其使用得到了 Springer Science and Business Media 的许可。

i.i.d. 样本

独立同分布样本，一组相互独立且服从同一分布的实例。

KDD

最初为“Knowledge Discovery from Databases”（基于数据库的知识发现）的缩写。如今广义上指“从数据中发现知识”，并常常被当作“数据挖掘”的同义词。

OLAP（MOLAP，ROLAP）

联机分析技术，通常与 MOLAP（多维 OLAP）同义。OLAP 引擎能促进多个（预先确定的）维度上的数据探索。OLAP 通常使用中间数据结构来存储预先计算的多维数据结果，从而提升计算效率。ROLAP（关系 OLAP）指用关系数据库执行 OLAP。

成本（效用 / 损失 / 回报）

当实际标签为 y 时，预测标签为 $\hat{y}$ 这一任务的成本（和 / 或收益）的度量指标。用准确率来评估模型时，需要假设错误的成本一致，而且分类正确的收益也一致。

错误率

参见准确率（错误率）。

分类器

从未标注实例到（离散）类的映射。分类器包括一种形式（如分类树）和一个解释程序（包括如何处理未知值等）。大部分分类器也能提供概率估计（或其他似然度评分）。可

以通过对其设置阈值来获得离散类决策，从而将成本效益或效用函数纳入考虑。

覆盖范围

分类器预测时所用的数据集的比例。如果分类器没有对所有实例进行分类，那么就需要知道它在有足够把握做出预测的实例集上的性能。

关联挖掘

一种挖掘技术，用于找到满足给定条件且形如“X 和 $Y \rightarrow A$ 和 B”（关联）的联合隐含规则。

归纳

通过一组数据构建一般模型（如分类树或方程）的过程。归纳与演绎相对：演绎根据一个一般规律或模型，以及一个或多个事实，来创造其他具体事实；而归纳从另一个方向入手，根据一系列事实创造一般规律或模型。在本书中，模型归纳与**学习**模型和**挖掘**模型同义，而且这些规律和模型通常都是统计性质的。

混淆矩阵

一个列出预测分类和实际分类的矩阵。混淆矩阵的大小是 $l \times l$，其中 l 为不同标签值的个数。多种分类器评估指标的定义均以混淆矩阵的内容为基础，包括准确率、真阳性比率、假阳性比率、真阴性比率、假阴性比率、精确度、召回率、敏感度、特异性、阳性预测值和阴性预测值。

机器学习

数据科学中，机器学习通常表示归纳算法在数据上的应用，往往与数据挖掘流程中的建模阶段同义混用。机器学习是科学研究的一个领域，关注归纳算法和其他可用于学习的算法。

记录

参见特征向量（记录，元组）。

交叉验证

通过将数据分为 k 个大小大致相同的互斥子集（“折叠”）来估计归纳器的准确率（或误差）的方法。归纳器要经过 k 次训练和测试，每次的训练集为 $k-1$ 个子集，而测试集为剩余的那个子集。交叉验证的准确率估计为 k 个折叠的准确率取平均，或组合（合并）后的测试折叠的准确率。

类（标签）

一个小型互斥标签集合，在分类问题中被用作目标变量的可能取值。标签数据中的每个数据项都有一个类标签，比如，美元钞票分类问题中的类分为**真钞**和**假钞**。股票评估问题中的类分为**飙升**、**暴跌**和**不变**。

敏感度

真阳性比率（见混淆矩阵）。

模式

对数据集属性及其性质的描述。

模型

能以描述或预测为目的，对一组数据进行概括或部分概括的结构和相应的解释。绝大多数归纳算法产生的模型能用作分类器、回归器、人类消费模式或下一步数据挖掘流程的输入。

模型部署

使用学习后的模型解决实际问题的过程。部署通常与数据挖掘流程的评估阶段中“使用”模型相对，后者中的部署通常是在答案已知的数据上模拟的。

缺失值

某属性的值未知或不存在的情形。值缺失的可能原因有很多，比如：没有测量值、仪器出现故障、属性不适用，或属性值无法得知。有些算法无法处理缺失值。

实例（示例，案例，记录）

用于模型学习或模型使用（如预测）的一个对象。在绝大多数数据科学工作中，实例由特征向量描述；有的数据科学工作则使用更复杂的表示方法（如包含实例间或实例的各部分之间的关系）。

示例

参见实例（示例，记录）。

属性（域，变量，特征）

描述一个实例的量。属性有一个由属性类型定义的域，而属性类型表示该属性可能的取值。常见的域有以下几种类型。

- **类别（符号）型域**
  可取有限个离散值。其中**标称型域**指变量值没有顺序，比如姓氏和颜色。而**序数型域**则指变量值之间存在顺序，比如取值为“低、中、高”的属性。
- **连续（数值）型域**
  通常为实数集的子集。不同的可能取值之间的差异可以被度量。实际问题中，整数经常被视为连续型变量。

  虽然本书中不会区分，但“特征”通常指的是属性的规范及其值。比如，颜色是一种属性，“颜色是蓝色”则是特征。许多对属性集的转化没有改变特征集（比如，重组属性值或把多值型属性转化为二值型属性）。本书与许多作者和从业者保持一致，把特征当作**属性**的同义词。

数据集

一个模式和符合该模式的一系列实例。一般认为，这些实例不必是有序的。绝大多数数据挖掘工作使用一个固定格式的表格，或一组特征向量。

数据清洗 / 清理

通过调整数据的形式或内容来提高数据质量的过程，比如删除或修正不正确的数据值。这一步骤通常在建模步骤之前，但经过整个数据挖掘过程后，可能会发现需要进一步的数据清洗，也可能会发现提升数据质量的方法。

数据挖掘

该术语含义丰富，有时指整个数据挖掘过程，有时指对数据应用具体的建模技术，以便构建模型或寻找其他模式 / 规律。

损失

参见成本（效用 / 损失 / 回报）。

特征

参见属性（域，变量，特征）。

特征向量（记录，元组）

描述一个实例的一列特征。

特异性

真阴性比率（见混淆矩阵）。

维度

一个或多个共同描述某性质的属性。比如，一个地理维度可能包含 3 个属性：国家、州、城市；而一个时间维度可能包含 5 个属性：年、月、日、时、分。

无监督学习

在没有预先指定目标属性的前提下，对实例进行分组的学习技术。聚类算法通常是无监督的。

先验

先验是从哲学中借用的术语，意指“先于经验”。在数据科学中，**先验信念**是问题中作为背景知识的信念。与之相对的是在检验数据之后形成的信念。你可以说“没有先验理由让我们相信该关系是线性的”。在检验数据后，你认为两个变量之间是线性关系（因而线性回归非常适用），但根据先验知识，你没有理由相信它们之间是线性关系。**先验**的反义词是**后验**。

效用

参见成本（效用 / 损失 / 回报）。

有监督学习

学习独立属性和指定的依赖属性（标签）之间关系的方法。绝大多数归纳算法属于有监督学习方法。

元组

参见特征向量（记录，元组）。

知识发现

辨别数据中有效、新颖、可能有用且最终可理解的模式的重要过程。以上定义出自 Fayyad Piatetsky-Shapiro, & Smyth（1996）的“知识发现和数据挖掘的进展”（Advances in Knowledge Discovery and Data Mining）。

域

参见属性。

准确率（错误率）

模型在数据集中预测正确（或错误）的比率。准确率通常基于未在任何阶段参与学习过程的独立（保留）数据集进行估计。更复杂的准确率估计技术，如交叉验证和自助法，也非常常用，对数据量较少的数据集而言尤其如此。

参考文献

[1] AAMODT, A., PLAZA, E. Case-based reasoning: foundational issues, methodological variations, and system approaches[J/OL]. Artificial intelligence communications, 1994, 7(1): 39–59. http://www.iiia.csic.es/~enric/AICom_ToC.html.

[2] ADAMS, N. M., HAND, D. J. Comparing classifiers when the misallocations costs are uncertain[J]. Pattern recognition, 1999, 32: 1139–1147.

[3] AHA, D. W. Lazy learning[M]. MA, USA: Kluwer Academic Publishers Norwell, 1997.

[4] AHA, D. W., KIBLER, D., ALBERT, M. K. Instance-based learning algorithms[J]. Machine learning, 1991, 6: 37–66.

[5] AGGARWAL, C., YU, P. Privacy-preserving data mining: models and algorithms[M]. New York City, USA: Springer, 2008.

[6] ARAL, S., MUCHNIK, L., SUNDARARAJAN, A. Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks[J]. Proceedings of the national academy of sciences, 2009, 106(51): 21544–21549.

[7] ARTHUR, D., VASSILVITSKII, S. K-means++: the advantages of careful seeding[C]. Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms, 1027–1035, 2007.

[8] ATTENBERG, J., IPEIROTIS, P., PROVOST, F. Beat the machine: challenging workers to find the unknown unknowns[C]. Workshops at the twenty-fifth AAAI conference on artificial intelligence, 2011.

[9] ATTENBERG, J., PROVOST, F. Why label when you can search: alternatives to active learning for applying human resources to build classification models under extreme class imbalance[C]. Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining, 423–432, 2010.

[10] BACHE, K., LICHMAN, M. UCI Machine learning repository[OL]. University of California. School of Information and Computer Science. [2013]. http://archive.ics.uci.edu/ml.

[11] BOLTON, R., HAND, D. Statistical fraud detection: a review[J]. Statistical science, 2002, 17(3): 235–255.

[12] BREIMAN, L., FRIEDMAN, J., OLSHEN, R., et al. Classification and regression trees[M]. Belmont, CA: Wadsworth International Group, 1984.

[13] BROOKS, D. What data can't do[N]. New York Times, 2013-02-18.

[14] BROWN, L., GANS, N., MANDELBAUM, A., et al. Statistical analysis of a telephone call center: a queueing-science perspective[J]. Journal of the American statistical association, 2005, 100(469): 36–50.

[15] BRYNJOLFSSON, E., SMITH, M. Frictionless commerce: a comparison of internet and conventional retailers[J]. Management science, 2000, 46: 563–585.

[16] BRYNJOLFSSON, E., HITT, L. M., KIM, H. H. Strength in numbers: how does datadriven decision making affect firm performance[OL]. [2011]. http://ssrn.com/abstract=1819486.

[17] Business insider. The Digital 100: the world's most valuable private tech companies[OL]. [2012]. http://www.businessinsider.com/2012-digital-100.

[18] CICCARELLI, F. D., DOERKS, T., VON MERING, C., et al. Toward automatic reconstruction of a highly resolved tree of life[J]. Science, 2006, 311(5765): 1283–1287.

[19] CLEARWATER, S., STERN, E. A rule-learning program in high energy physics event classification[J]. Comp physics comm, 1991, 67: 159–182.

[20] CLEMONS, E., THATCHER, M. Capital One: exploiting and information-based strategy[C]. Proceedings of the 31st Hawaii international conference on system sciences, 1998.

[21] COHEN, L., DIETHER, K., MALLOY, C. Legislating stock prices[D]. Boston: Harvard Business School, 2012.

[22] COVER, T., HART, P. Nearest neighbor pattern classification[J]. Information theory, IEEE transactions on, 1967, 13(1): 21–27.

[23] CRANDALL, D., BACKSTROM, L., COSLEY, D., et al. Inferring social ties from geographic coincidences[J]. Proceedings of the national academy of sciences, 2010, 107(52): 22436–22441.

[24] DEZA, E., DEZA, M. Dictionary of distances[M]. Elsevier science, 2006.

[25] DIETTERICH, T. G. Approximate statistical tests for comparing supervised classification learning algorithms[J]. Neural computation, 1998, 10: 1895–1923.

[26] DIETTERICH, T. G. Ensemble methods in machine learning[C]. Multiple classifier systems, 2000, 1–15.

[27] DUHIGG, C. How companies learn your Secrets[N]. New York Times, 2012-02-19.

[28] ELMAGARMID, A., IPEIROTIS, P., VERYKIOS, V. Duplicate record detection: a survey[J]. Knowledge and data engineering, IEEE transactions on, 2007, 19(1): 1–16.

[29] EVANS, R., FISHER, D. Using decision tree induction to minimize process delays in the printing industry[M]//Handbook of data mining and knowledge discovery, 874–881. Oxford: Oxford University Press, 2002.

[30] EZAWA, K., SINGH, M., NORTON, S. Learning goal oriented Bayesian networks for telecommunications risk management[C]. Proceedings of the thirteenth international conference on machine learning, 139–147. San Francisco, CA: Morgan Kaufmann, 1996.

[31] FAWCETT, T. An introduction to ROC analysis[J]. Pattern recognition letters, 2006, 27(8): 861–874.

[32] FAWCETT, T., PROVOST, F. Combining data mining and machine learning for effective user profiling[C]. Proceedings of the second international conference on knowledge discovery and data mining, 8–13. CA: AAAI Press Menlo Park, 1996.

[33] FAWCETT, T., PROVOST, F. Adaptive fraud detection[J]. Data mining and knowledge discovery, 1997, 1(3): 291–316.

[34] FAYYAD, U., PIATETSKY-SHAPIRO, G., SMYTH, P. From data mining to knowledge discovery in databases[J]. AI magazine, 1996, 17: 37–54.

[35] FRANK, A., ASUNCION, A. UCI machine learning repository[OL]. University of California. School of Information and Computer Science. [2010]. http://archive.ics.uci.edu/ml.

[36] FRIEDMAN, J. On bias, variance, 0/1-loss, and the curse-of-dimensionality[J]. Data mining and knowledge discovery, 1997, 1(1): 55–77.

[37] GANDY, O. H. Coming to terms with chance: engaging rational discrimination and cumulative disadvantage[M]. USA: Ashgate Publishing Company, 2009.

[38] GOLDFARB, A. TUCKER, C. Online advertising, behavioral targeting, and privacy. Communications of the ACM, 2011, 54(5): 25–27.

[39] HAIMOWITZ, I., SCHWARTZ, H. Clustering and prediction for credit line optimization[C]. AI approaches to fraud detection and risk management, 29–33. Palo Alto, CA, USA: AAAI Press, 1997.

[40] HALL, M., FRANK, E., HOLMES, G., et al. The WEKA data mining software: an update[J]. SIGKDD explorations, 2009, 11(1).

[41] HAND, D. J. Statistics: a very short introduction[M]. Oxford, UK: Oxford University Press, 2008.

[42] HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. The elements of statistical learning: data mining, inference, and prediction[M]. 2nd ed. New York City, USA: Springer, 2009.

[43] HAYS, C. L. What they know about you[N]. The New York Times, 2004-11-14.

[44] HERNANDEZ, M. A., STOLFO, S. J. The merge/purge problem for large databases[J]. SIGMOD Rec., 1995, 24: 127–138.

[45] HILL, S., PROVOST, F., VOLINSKY, C. Network-based marketing: identifying likely adopters via consumer networks. Statistical science, 2006, 21(2): 256–276.

[46] Holte, R. C. Very simple classification rules perform well on most commonly used datasets[J]. Machine learning, 1993, 11: 63–91.

[47] IPEIROTIS, P., PROVOST, F., WANG, J. Quality management on Amazon Mechanical Turk[C]. Proceedings of the 2010 ACM SIGKDD workshop on human computation, 64–67, 2010.

[48] JACKSON, M. Michael Jackson's malt whisky companion: a connoisseur's guide to the malt whiskies of Scotland[M]. London: Dorling Kindersley, 1989.

[49] JAPKOWICZ, N., STEPHEN, S. The class imbalance problem: a systematic study[J]. Intelligent data analysis, 2002, 6(5): 429–450.

[50] JAPKOWICZ, N., SHAH, M. Evaluating learning algorithms: a classification perspective[M]. Cambridge UK: Cambridge University Press, 2011.

[51] JENSEN, D. D., COHEN, P. R. Multiple comparisons in induction algorithms[J]. Machine learning, 2000, 38(3): 309–338.

[52] JUNQUE DE FORTUNY, E., MARTENS, D., PROVOST, F. Predictive modeling with big data: is bigger really better?[J/OL] Big data. [2013-10]. http://online.liebertpub.com/doi/abs/10.1089/big.2013.0037.

[53] KASS, G. V. An exploratory technique for investigating large quantities of categorical data[J]. Applied statistics, 1980, 29(2): 119–127.

[54] KAUFMAN, S., ROSSET, S., PERLICH, C., et al. Leakage in data mining: formulation, detection, and avoidance[J]. ACM Transactions on knowledge discovery from data, 2012, 6(4): 15.

[55] KOHAVI, R., BRODLEY, C., FRASCA, B., et al. KDD-cup 2000 organizers' report: peeling the onion[C]. ACM SIGKDD explorations, 2000, 2(2).

[56] KOHAVI, R., DENG, A., FRASCA, B., et al. Trustworthy online controlled experiments: five puzzling outcomes explained[C]. Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining, 2012, 786–794.

[57] KOHAVI, R., LONGBOTHAM, R. Online experiments: lessons learned[J]. Computer, 2007, 40(9): 103–105.

[58] KOHAVI, R., LONGBOTHAM, R., SOMMERFIELD, D., et al. Controlled experiments on the web: survey and practical guide[J]. Data mining and knowledge discovery, 2009, 18(1): 140–181.

[59] KOHAVI, R., PAREKH, R. Ten supplementary analyses to improve e-commerce web sites[C]. Proceedings of the fifth WEBKDD workshop, 2003.

[60] KOHAVI, R., PROVOST, F. Glossary of terms[J]. Machine learning, 1998, 30(2-3): 271–274.

[61] KOLODNER, J. Case-based reasoning[M]. San Mateo, CA, USA: Morgan Kaufmann, 1993.

[62] KOREN, Y., BELL, R., VOLINSKY, C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8): 30–37.

[63] KOSINSKI, M., STILLWELL, D., GRAEPEL, T. Private traits and attributes are predictable from digital records of human behavior[C]. Proceedings of the national academy of sciences, 2013.

[64] LAPOINTE, F.-J., LEGENDRE, P. A classification of pure malt Scotch whiskies[J]. Applied statistics, 1994, 43(1): 237–257.

[65] LEIGH, D. Neural networks for credit scoring[M]. Intelligent systems for finance and business, 61–69. West Sussex, England: John Wiley and Sons Ltd., 1995.

[66] LETUNIC, BORK. Interactive tree of life (iTOL): an online tool for phylogenetic tree display and annotation[J]. Bioinformatics, 23(1), 2006.

[67] LIN, J.-H., VITTER, J. S. A theory for memory-based learning[J]. Machine learning, 1994, 17: 143–167.

[68] LLOYD, S. P. Least square quantization in PCM[J]. IEEE transactions on information theory, 1982, 28(2): 129–137.

[69] MACKAY, D. An example inference task: clustering[M]//Information theory, inference and learning algorithms. Cambridge, UK: Cambridge University Press, 2003.

[70] MACQUEEN, J. B. Some methods for classification and analysis of multivariate observations[C]. Proceedings of 5th berkeley symposium on mathematical statistics and probability, 281–297, 1967. Oakland, CA, USA: University of California Press.

[71] MALIN, B., SWEENEY, L. How (not) to protect genomic data privacy in a distributed network: using trail re-identification to evaluate and design anonymity protection systems[J]. Journal of biomedical informatics, 2004, 37(3): 179–192.

[72] MARTENS, D., PROVOST, F. Pseudo-social network targeting from consumer transaction data[D]. New York University. Stern School of Business, 2011.

[73] MCCALLUM, A., NIGAM, K. A comparison of event models for naive Bayes text classification[C]. AAAI workshop on learning for text categorization, 1988.

[74] MCDOWELL, G. Cracking the coding interview: 150 programming questions and solutions[R]. CareerCup LLC, 2008.

[75] MCNAMEE, M. Credit card revolutionary[J]. Stanford Business, 2001, 69(3).

[76] MCPHERSON, M., SMITH-LOVIN, L., COOK, J. M. Birds of a feather: homophily in social networks[J]. Annual review of sociology, 2001, 27: 415–444.

[77] MITTERMAYER, M., KNOLMAYER, G. Text mining systems for market response to news: a survey[D]. University of Bern. Institute of Information Systems, 2006.

[78] MUOIO, A. They have a better idea ... do you?[J]. Fast company, 1997, 10.

[79] NISSENBAUM, H. Privacy in context[M]. Palo Alto, CA, USA: Stanford University Press, 2010.

[80] PAPADOPOULOS, A. N., MANOLOPOULOS, Y. Nearest neighbor search: a database perspective[M]. New York City, USA: Springer, 2005.

[81] PENNISI, E. A tree of life[OL]. [2003]. http://www.sciencemag.org/site/feature/data/tol/.

[82] PERLICH, C., PROVOST, F., SIMONOFF, J. Tree induction vs. logistic regression: a learning-curve analysis[J]. Journal of machine learning research, 2003, 4: 211–255.

[83] PERLICH, C., DALESSANDRO, B., STITELMAN, O., et al. Machine learning for targeted display advertising: transfer learning in action[J]. Machine learning, 2013.

[84] POUNDSTONE, W. Are you smart enough to work at google: trick questions, zen-like riddles, insanely difficult puzzles, and other devious interviewing techniques you need to know to get a job anywhere in the new economy[M]. New York City, USA: Little, Brown and Company, 2012.

[85] PROVOST, F., FAWCETT, T. Analysis and visualization of classifier performance: comparison under imprecise class and cost distributions[C]. Proceedings of the third international conference on knowledge discovery and data mining (KDD-97), 43–48. Menlo Park, CA, USA: AAAI Press, 1997.

[86] PROVOST, F., FAWCETT, T. Robust classification for imprecise environments[J]. Machine learning, 2001, 42(3): 203–231.

[87] PROVOST, F., FAWCETT, T., Kohavi, R. The case against accuracy estimation for comparing induction algorithms[C]. Proceedings of ICML-98, 445–453. San Francisco, CA, USA: Morgan Kaufmann, 1998.

[88] PYLE, D. Data preparation for data mining[M]. San Francisco, CA, USA: Morgan Kaufmann, 1999.

[89] QUINE, W.V.O. Two dogmas of empiricism[J]. The philosophical review, 1951, 60: 20–43.

[90] QUINLAN, J. R. C4.5: Programs for machine learning[M]. San Francisco, CA, USA: Morgan Kaufmann, 1993.

[91] QUINLAN, J. Induction of decision trees[J]. Machine learning, 1986, 1(1): 81–106.

[92] RAEDER, T., Dalessandro, B., Stitelman, O. Design principles of massive, robust prediction systems[C]. Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining, 2012.

[93] ROSSET, S., ZHU, J. Piecewise linear regularized solution paths[J]. The annals of statistics, 2007, 35(3): 1012–1030.

[94] SCHUMAKER, R., CHEN, H. A discrete stock price prediction engine based on financial news keywords[J]. IEEE computer, 2010, 43(1): 51–56.

[95] SENGUPTA, S. Facebook's prospects may rest on trove of data[N]. New York Times 2012-05-15.

[96] SHAKHNAROVICH, G., DARRELL, T., INDYK, P. Nearest-neighbor methods in learning and vision[M]. Cambridge, MA, USA: The MIT Press, 2005.

[97] SHANNON, C. E. A mathematical theory of communication[J]. Bell system technical journal, 1948, 27: 379–423.

[98] SHEARER, C. The CRISP-DM model: the new blueprint for data mining[J]. Journal of data war☺ousing, 2000, 5(4): 13–22.

[99] SHMUELI, G. To explain or to predict[J]. Statistical science, 2010, 25(3): 289–310.

[100] SILVER, N. The signal and the noise[M]. London, UK: The Penguin Press HC, 2012.

[101] SOLOVE, D. A taxonomy of privacy[J]. University of Pennsylvania law review, 2006, 154(3): 477–564.

[102] STEIN, R. M. The relationship between default prediction and lending profits: integrating ROC analysis and loan pricing[J]. Journal of banking and finance, 2005, 29: 1213–1236.

[103] SUGDEN, A. M., JASNY, B. R., CULOTTA, E. Charting the evolutionary history of life[J]. Science, 2003, 300(5626).

[104] SWETS, J. A. Measuring the accuracy of diagnostic systems[J]. Science, 1988, 240, 1285–1293.

[105] SWETS, J. A. Signal detection theory and ROC analysis in psychology and diagnostics: collected papers[M]. Mahwah, NJ, USA, Lawrence Erlbaum Associates, 1996.

[106] SWETS, J. A., DAWES, R. M., MONAHAN, J. Better decisions through science[J]. Scientific american, 2000, 283: 82–87.

[107] TAMBE, P. Big data investment, skills, and firm value[OL]. New York University. Stern School of Bussiness. 2013. http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2294077.

[108] WEKA. Weka machine learning software[OL]. 2001. http://www.cs.waikato.ac.nz/~ml/index.html.

[109] Wikipedia. Determining the number of clusters in a data set[OL]. Wikipedia, the free encyclopedia. (2013-02-14). http://en.wikipedia.org/wiki/Determining_the_number_of_clusters_in_a_data_set.

[110] WILCOXON, F. Individual comparisons by ranking methods[J/OL]. Biometrics bulletin, 1945, 1(6): 80–83. http://sci2s.ugr.es/keel/pdf/algorithm/articulo/wilcoxon1945.pdf.

[111] Winterberry group beyond the grey areas: transparency, brand safety and the future of online advertising[R/OL].Winterberry Group LLC, 2010. http://www.winterberrygroup.com/ourinsights/wp.

[112] WISHART, D. Whisky classified: choosing single malts by flavour[M]. London UK: Pavilion, 2006.

[113] WITTEN, I., FRANK, E. Data mining: practical machine learning tools and techniques with Java implementations[M/OL]. San Francisco, CA, USA: Morgan Kaufmann, 2000. http://www.cs.waikato.ac.nz/~ml/weka/.

[114] ZADROZNY, B. Learning and evaluating classifiers under sample selection bias[C]. Proceedings of the twenty-first international conference on machine learning, 903–910, 2004.

[115] ZADROZNY, B., ELKAN, C. Learning and making decisions when costs and probabilities are both unknown[C]. Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining. 204–213, 2001.

关于作者

福斯特 • 普罗沃斯特（**Foster Provost**），纽约大学斯特恩商学院教授、NEC 教员，讲授商业分析、数据科学和 MBA 课程，其获奖研究被广泛阅读和引用。在执教纽约大学前，他曾作为研究型数据科学家为 Verizon 公司的前身工作五年。过去十年，他还参与创立了数家由数据科学驱动的成功企业。

汤姆 • 福西特（**Tom Fawcett**），机器学习博士，已在业界研发领域工作二十余年，曾进入 GTE 实验室、NYNEX/Verizon 实验室和 HP 实验室等机构。他发表的文章已经成为数据科学领域中方法论（如评估数据挖掘结果）和应用（如欺诈检测和垃圾邮件过滤）方面的范文。

O'REILLY®

商战数据挖掘：你需要了解的数据科学与分析思维

在现代社会中，数据即商业，它是提升生产力、促进创新和获取用户洞见的基础，数据思维和分析方法可谓是新时代的商战孙子兵法，只有善用数据者才能在这个数据驱动的环境中获得竞争优势。本书通过大量真实的商业问题案例，介绍数据科学的基本原理和各种数据挖掘技术，阐释如何从数据中提取出有用信息，进而用数据科学方法解决商业问题，做出精准的决策。

“对于每一个真诚拥抱大数据机遇的人来说，这都是一本不可不读之书。”

——Craig Vaughan

SAP全球副总裁

“这本书与众不同，因为它没有详解算法，而是帮读者理解数据科学背后的基本概念，最重要的是，它指导读者如何着手解决问题并取得成功。无论是想综合了解数据科学的普通人，还是需要学习基础知识的数据科学从业者，都要读一读这本书。”

——Chris Volinsky

AT&T实验室统计研究主管

“这本书远不止是数据分析入门书，对所有需要做出数据驱动型决策的人来说，这本书不容错过。”

——Tom Phillips

Dstillery首席执行官，Google搜索和分析前主管

“这是一本通俗易懂的入门读物，既能帮助商务人士更好地领会数据科学家所用的概念、工具和技术，又能帮助数据科学家更好地理解其解决方案所应用的商业背景。”

——Joe McCarthy

Atigeo分析与数据科学主管

封面设计：Mark Paglietti 张健

图灵社区：iTuring.cn
热线：(010)51095183转600

分类建议 计算机／数据科学

人民邮电出版社网址：www.ptpress.com.cn

O'Reilly Media, Inc.授权人民邮电出版社出版





ISBN 978-7-115-52233-7
定价：89.00元

看完了

如果您对本书内容有疑问，可发邮件至 contact@turingbook.com，会有编辑或作译者协助答疑。也可访问图灵社区，参与本书讨论。

如果是有关电子书的建议或问题，请联系专用客服邮箱:
ebook@turingbook.com。

在这可以找到我们:

微博 @图灵教育：好书、活动每日播报
微博 @图灵社区：电子书和好文章的消息
微博 @图灵新知：图灵教育的科普小组
微信 图灵访谈：ituring_interview，讲述码农精彩人生
微信 图灵教育：turingbooks